



# System Event Log Troubleshooting Guide for Intel® S5500/S3420 Series Server Boards

*Intel order number G74211-002*



**Revision 1.1**

**December 2013**

**Platform Collaboration and Systems Division – Marketing**

## ***Revision History***

Date	Revision Number	Modifications
August 2012	1.0	Initial draft.
December 2013	1.1	<ul style="list-style-type: none"><li>▪ Corrected IPMI Watchdog and PEF Sensors Typical Characteristics tables.</li><li>▪ Clarified Channel designators for DIMM memory errors.</li><li>▪ Added ME sensor 17h.</li></ul>

## ***Disclaimers***

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: <http://www.intel.com/design/literature>.

## ***Table of Contents***

<b>1. Introduction .....</b>	<b>1</b>
1.1 Purpose.....	1
1.2 Industry Standard.....	1
1.2.1 Intelligent Platform Management Interface (IPMI).....	1
1.2.2 Baseboard Management Controller (BMC).....	2
1.2.3 Intel® Intelligent Power Node Manager Version 1.5 .....	3
<b>2. Basic Decoding of a SEL Record .....</b>	<b>4</b>
2.1 Default Values in the SEL Records .....	4
<b>3. Sensor Cross Reference List.....</b>	<b>8</b>
3.1 BMC owned Sensors (GID = 0020h) .....	8
3.2 BIOS POST owned Sensors (GID = 0001h) .....	12
3.3 BIOS SMI owned Sensors (GID = 0033h) .....	12
3.4 Hot Swap Controller Firmware owned Sensors (GID = 00C0h/00C2h).....	14
3.5 Node Manager / ME Firmware owned Sensors (GID = 002Ch or 602Ch).....	15
3.6 Microsoft* OS owned Events (GID = 0041) .....	16
3.7 Linux* Kernel Panic Events (GID = 0021).....	16
<b>4. Power Subsystems.....</b>	<b>17</b>
4.1 Voltage Sensors.....	17
4.2 Power Unit .....	21
4.2.1 Power Unit Status Sensor .....	21
4.2.2 Power Unit Redundancy Sensor.....	22
4.3 Power Supply .....	24
4.3.1 Power Supply Status Sensors .....	24
4.3.2 Power Supply AC Power Input Sensors .....	25
4.3.3 Power Supply Current Output % Sensors.....	26
4.3.4 Power Supply Temperature Sensors.....	27
<b>5. Cooling Subsystem .....</b>	<b>29</b>
5.1 Fan Sensors.....	29
5.1.1 Fan Speed Sensors.....	29
5.1.2 Fan Presence and Redundancy Sensors .....	30
5.2 Temperature Sensors.....	33
5.2.1 Regular Temperature Sensors .....	33
5.2.2 Thermal Margin Sensors .....	35
5.2.3 Processor Thermal Control % Sensors.....	36
5.2.4 Discrete Thermal Sensors .....	37
<b>6. Processor Subsystem .....</b>	<b>39</b>
6.1 Processor Status Sensor.....	39
6.2 Catastrophic Error Sensor .....	40
6.2.1 Catastrophic Error Sensor – Next Steps.....	41

6.3	CPU Missing Sensor .....	41
6.3.1	CPU Missing Sensor – Next Steps .....	42
6.4	QuickPath Interconnect Error Sensors .....	42
6.4.1	QPI Correctable Error Sensor .....	42
6.4.2	QPI Non-Fatal Error Sensor .....	43
6.4.3	QPI Fatal and Fatal #2 .....	44
<b>7.</b>	<b>Memory Subsystem.....</b>	<b>46</b>
7.1	Memory RAS Mirroring and Sparing .....	46
7.1.1	Mirroring Configuration Status .....	46
7.1.2	Mirrored Redundancy State Sensor .....	47
7.1.3	Sparing Configuration Status.....	49
7.1.4	Sparing Redundancy State Sensor .....	50
7.2	ECC and Address Parity .....	53
7.2.1	Memory Correctable and Uncorrectable ECC Error .....	53
7.2.2	Memory Address Parity Error .....	55
<b>8.</b>	<b>PCI Express* and Legacy PCI Subsystem .....</b>	<b>58</b>
8.1	PCI Express* Errors .....	58
8.1.1	PCI Express* Correctable Errors .....	58
8.1.2	PCI Express* Fatal Errors .....	59
8.1.3	Legacy PCI Errors.....	61
<b>9.</b>	<b>System BIOS Events .....</b>	<b>63</b>
9.1	System Events .....	63
9.1.1	System Boot.....	63
9.1.2	Timestamp Clock Synchronization .....	63
9.2	System Firmware Progress (Formerly Post Error) .....	64
9.2.1	System Firmware Progress (Formerly Post Error) – Next Steps .....	65
<b>10.</b>	<b>Chassis Subsystem.....</b>	<b>71</b>
10.1	Physical Security .....	71
10.1.1	Chassis Intrusion.....	71
10.1.2	LAN Leash Lost.....	71
10.2	FP (NMI) Interrupt .....	73
10.2.1	FP (NMI) Interrupt – Next Steps .....	73
10.3	Button Press Events.....	74
<b>11.</b>	<b>Miscellaneous Events .....</b>	<b>75</b>
11.1	IPMI Watchdog.....	75
11.2	SMI Timeout.....	77
11.2.1	SMI Timeout – Next Steps.....	77
11.3	System Event Log Cleared.....	78
11.4	System Event – PEF Action .....	78
11.4.1	System Event – PEF Action – Next Steps .....	79
<b>12.</b>	<b>Hot Swap Controller Events.....</b>	<b>80</b>

12.1	HSC Backplane Temperature Sensor .....	80
12.2	HSC Drive Slot Status Sensor .....	81
12.2.1	HSC Drive Slot Status Sensor – Next Steps .....	82
12.3	HSC Drive Presence Sensor .....	82
12.3.1	HSC Drive Presence Sensor – Next Steps .....	83
<b>13.</b>	<b>Manageability Engine (ME) Events .....</b>	<b>85</b>
13.1	Node Manager Exception Event .....	85
13.1.1	Node Manager Exception Event – Next Steps .....	86
13.2	Node Manager Health Event .....	86
13.2.1	Node Manager Health Event – Next Steps .....	87
13.3	Node Manager Operational Capabilities Change .....	88
13.3.1	Node Manager Operational Capabilities Change – Next Steps .....	89
13.4	Node Manager Alert Threshold Exceeded .....	90
13.4.1	Node Manager Alert Threshold Exceeded – Next Steps .....	91
13.5	ME Firmware Health Event .....	91
13.5.1	ME Firmware Health Event – Next Steps .....	92
<b>14.</b>	<b>Microsoft Windows* Records .....</b>	<b>93</b>
14.1	Boot-up Event Records .....	93
14.2	Shutdown Event Records .....	94
14.3	Bug Check / Blue Screen Event Records .....	97
<b>15.</b>	<b>Linux* Kernel Panic Records .....</b>	<b>99</b>

## ***List of Tables***

Table 1: SEL Record Format.....	4
Table 2: Event Request Message Event Data Field Contents .....	6
Table 3: OEM SEL Record (Type C0h-DFh) .....	7
Table 4: OEM SEL Record (Type E0h-FFh).....	7
Table 5: BMC owned Sensors.....	8
Table 6: BIOS POST owned Sensors .....	12
Table 7: BIOS SMI owned Sensors.....	13
Table 8: Hot Swap Controller Firmware owned Sensors .....	14
Table 9: Management Engine Firmware owned Sensors .....	15
Table 10: Microsoft® OS owned Events .....	16
Table 11: Linux® Kernel Panic Events .....	16
Table 12: Voltage Sensors Typical Characteristics .....	17
Table 13: Voltage Sensors Event Triggers – Description .....	18
Table 14: Voltage Sensors – Next Steps.....	18
Table 15: Power Unit Status Sensors Typical Characteristics .....	21
Table 16: Power Unit Status Sensor – Sensor Specific Offsets – Next Steps.....	22
Table 17: Power Unit Redundancy Sensors Typical Characteristics .....	23
Table 18: Power Unit Redundancy Sensor – Event Trigger Offset – Next Steps.....	23
Table 19: Power Supply Status Sensors Typical Characteristics.....	24
Table 20: Power Supply Status Sensor – Sensor Specific Offsets – Next Steps .....	24
Table 21: Power Supply AC Power Input Sensors Typical Characteristics .....	25
Table 22: Power Supply AC Power Input Sensor – Event Trigger Offset – Next Steps.....	26
Table 23: Power Supply Current Output % Sensors Typical Characteristics .....	26
Table 24: Power Supply Current Output % Sensor – Event Trigger Offset – Next Steps.....	27
Table 25: Power Supply Temperature Sensors Typical Characteristics .....	28
Table 26: Power Supply Temperature Sensor – Event Trigger Offset – Next Steps .....	28
Table 27: Fan Speed Sensors Typical Characteristics .....	29
Table 28: Fan Speed Sensor – Event Trigger Offset – Next Steps.....	30
Table 29: Fan Presence Sensors Typical Characteristics .....	30
Table 30: Fan Presence Sensors – Event Trigger Offset – Next Steps .....	31
Table 31: Fan Redundancy Sensors Typical Characteristics.....	31
Table 32: Fan Redundancy Sensor – Event Trigger Offset – Next Steps .....	32
Table 33: Temperature Sensors Typical Characteristics .....	33
Table 34: Temperature Sensors Event Triggers – Description .....	34
Table 35: Temperature Sensors – Next Steps.....	34
Table 36: Thermal Margin Sensors Typical Characteristics.....	35
Table 37: Thermal Margin Sensors Event Triggers – Description.....	35
Table 38: Thermal Margin Sensors – Next Steps .....	36
Table 39: Processor Thermal Control % Sensors Typical Characteristics .....	36

Table 40: Processor Thermal Control % Sensors Event Triggers – Description .....	37
Table 41: Processor Thermal Control % Sensors – Next Steps .....	37
Table 42: Discrete Thermal Sensors Typical Characteristics.....	38
Table 43: Discrete Thermal Sensors – Next Steps.....	38
Table 44: Process Status Sensors Typical Characteristics.....	39
Table 45: Processor Status Sensors – Next Steps.....	40
Table 46: Catastrophic Error Sensor Typical Characteristics.....	40
Table 47: CPU Missing Sensor Typical Characteristics.....	41
Table 48: QPI Correctable Error Sensor Typical Characteristics .....	42
Table 49: QPI Non-Fatal Error Sensor Typical Characteristics.....	43
Table 50: QPI Fatal Error Sensor Typical Characteristics .....	44
Table 51: QPI Fatal #2 Error Sensor Typical Characteristics.....	45
Table 52: Mirroring Configuration Status Sensor Typical Characteristics .....	46
Table 53: Mirroring Configuration Status Sensor Event Trigger Offset – Next Steps .....	47
Table 54: Mirrored Redundancy State Sensor Typical Characteristics .....	48
Table 55: Mirrored Redundancy State Sensor Event Trigger Offset – Next Steps.....	49
Table 56: Sparing Configuration Status Sensor Typical Characteristics .....	49
Table 57: Sparing Configuration Status Sensor Event Trigger Offset – Next Steps.....	50
Table 58: Sparing Redundancy State Sensor Typical Characteristics .....	51
Table 59: Sparing Redundancy State Sensor Event Trigger Offset – Next Steps.....	52
Table 60: Correctable and Uncorrectable ECC Error Sensor Typical Characteristics .....	53
Table 61: Correctable and Uncorrectable ECC Error Sensor Event Trigger Offset – Next Steps.....	54
Table 62: Address Parity Error Sensor Typical Characteristics .....	55
Table 63: PCI Express* Correctable Error Sensor Typical Characteristics .....	58
Table 64: PCI Express* Correctable Error Sensor Event Trigger Offset – Next Steps.....	59
Table 65: PCI Express* Fatal Error Sensor Typical Characteristics.....	60
Table 66: PCI Express* Fatal Error Sensor Event Trigger Offset – Next Steps .....	60
Table 67: Legacy PCI Error Sensor Typical Characteristics .....	62
Table 68: Legacy PCI Error Sensor Event Trigger Offset – Next Steps.....	62
Table 69: System Event Sensor Typical Characteristics .....	64
Table 70: POST Error Sensor Typical Characteristics.....	65
Table 71: POST Error Codes .....	65
Table 72: Physical Security Sensor Typical Characteristics .....	71
Table 73: Physical Security Sensor Event Trigger Offset – Next Steps .....	72
Table 74: FP (NMI) Interrupt Sensor Typical Characteristics.....	73
Table 75: Button Press Events Sensor Typical Characteristics .....	74
Table 76: IPMI Watchdog Sensor Typical Characteristics .....	75
Table 77: IPMI Watchdog Sensor Event Trigger Offset – Next Steps.....	76
Table 78: SMI Timeout Sensor Typical Characteristics .....	77
Table 79: System Event Log Cleared Sensor Typical Characteristics .....	78
Table 80: System Event – PEF Action Sensor Typical Characteristics.....	79



Table 81: HSC Backplane Temperature Sensor Typical Characteristics .....	80
Table 82: HSC Backplane Temperature Sensor – Event Trigger Offset – Next Steps .....	81
Table 83: HSC Drive Slot Status Sensor Typical Characteristics .....	81
Table 84: HSC Drive Presence Sensor Typical Characteristics.....	83
Table 85: Node Manager Exception Sensor Typical Characteristics .....	85
Table 86: Node Manager Health Event Sensor Typical Characteristics.....	86
Table 87: Node Manager Operational Capabilities Change Sensor Typical Characteristics .....	88
Table 88: Node Manager Alert Threshold Exceeded Sensor Typical Characteristics .....	90
Table 89: ME Firmware Health Event Sensor Typical Characteristics .....	91
Table 90: ME Firmware Health Event Sensor – Next Steps .....	92
Table 91: Boot-up Event Record Typical Characteristics.....	93
Table 92: Boot-up OEM Event Record Typical Characteristics.....	94
Table 93: Shutdown Reason Code Event Record Typical Characteristics.....	95
Table 94: Shutdown Reason OEM Event Record Typical Characteristics .....	95
Table 95: Shutdown Comment OEM Event Record Typical Characteristics .....	96
Table 96: Bug Check / Blue Screen – OS Stop Event Record Typical Characteristics .....	97
Table 97: Bug Check / Blue Screen Code OEM Event Record Typical Characteristics .....	97
Table 98: Linux* Kernel Panic Event Record Characteristics .....	99
Table 99: Linux* Kernel Panic String Extended Record Characteristics .....	100



# 1. Introduction

---

The server management hardware that is part of Intel® Server Boards and Intel® Server Platforms serves as a vital part of the overall server management strategy. The server management hardware provides essential information to the system administrator and provides the administrator the ability to remotely control the server, even when the operating system is not running.

The Intel® Server Boards and Intel® Server Platforms offer comprehensive hardware and software based solutions. The server management features make the servers simple to manage and provide alerting on system events. From entry to enterprise systems, good overall server management is essential to reduce overall total cost of ownership.

This *Troubleshooting Guide* is intended to help the users better understand the events that are logged in the Baseboard Management Controllers (BMC) System Event Logs (SEL) on these Intel® Server Boards.

There is a separate *User's Guide* that covers the general server management and the server management software offered on Intel® Server Boards and Intel® Server Platforms.

Server boards currently supported by this document:

- Intel® S3200/X38ML Server Boards
- Intel® S5500/S3420 Series Server Boards

## 1.1 Purpose

The purpose of this document is to list all possible events generated by the Intel® platform. It may be possible that other sources (not under our control) also generate events, which will not be described in this document.

## 1.2 Industry Standard

### 1.2.1 Intelligent Platform Management Interface (IPMI)

The key characteristic of the Intelligent Platform Management Interface (IPMI) is that the inventory, monitoring, logging, and recovery control functions are available independently of the main processors, BIOS, and operating system. Platform management functions can also be made available when the system is in a power-down state.

IPMI works by interfacing with the BMC, which extends management capabilities in the server system and operates independently of the main processor by monitoring the on-board instrumentation. Through the BMC, IPMI also allows administrators to control power to the server, and remotely access BIOS configuration and operating system console information.

IPMI defines a common platform instrumentation interface to enable interoperability between:

- The baseboard management controller and chassis
- The baseboard management controller and systems management software
- Between servers

IPMI enables the following:

- Common access to platform management information, consisting of:
  - Local access from systems management software
  - Remote access from LAN
  - Inter-chassis access from Intelligent Chassis Management Bus
  - Access from LAN, serial/modem, IPMB, PCI SMBus\*, or ICMB, available even if the processor is down
- IPMI interface isolates systems management software from hardware.
- Hardware advancements can be made without impacting the systems management software.
- IPMI facilitates cross-platform management software.

You can find more information on IPMI at the following URL:

<http://www.intel.com/design/servers/ipmi>

## 1.2.2 Baseboard Management Controller (BMC)

A baseboard management controller (BMC) is a specialized microcontroller embedded on most Intel® Server Boards. The BMC is the heart of the IPMI architecture and provides the intelligence behind intelligent platform management, that is, the autonomous monitoring and recovery features implemented directly in platform management hardware and firmware.

Different types of sensors built into the computer system report to the BMC on parameters such as temperature, cooling fan speeds, power mode, operating system status, and so on. The BMC monitors the system for critical events by communicating with various sensors on the system board; it sends alerts and logs events when certain parameters exceed their preset thresholds, indicating a potential failure of the system. The administrator can also remotely communicate with the BMC to take some corrective actions such as resetting or power cycling the system to get a hung OS running again. These abilities save on the total cost of ownership of a system.

For Intel® Server Boards and Intel® Server Platforms, the BMC supports the industry-standard *IPMI 2.0 Specification*, enabling you to configure, monitor, and recover systems remotely.

### 1.2.2.1 System Event Log (SEL)

The BMC provides a centralized, non-volatile repository for critical, warning, and informational system events called the System Event Log or SEL. By having the BMC manage the SEL and logging functions, it helps to ensure that “post-mortem” logging information is available if a failure occurs that disables the systems processor(s).

The BMC allows access to SEL from in-band and out-of-band mechanisms. There are various tools and utilities that can be used to access the SEL. There is the Intel® SELViewer and multiple open sourced IPMI tools.

### 1.2.3 Intel® Intelligent Power Node Manager Version 1.5

Intel® Intelligent Power Node Manager version 1.5 (NM) is a platform-resident technology that enforces power and thermal policies for the platform. These policies are applied by exploiting subsystem knobs (such as processor P and T states) that can be used to control power consumption. Intel® Intelligent Power Node Manager enables data center power and thermal management by exposing an external interface to management software through which platform policies can be specified. It also enables specific data center power management usage models such as power limiting.

The configuration and control commands are used by the external management software or BMC to configure and control the Intel® Intelligent Power Node Manager feature. Because Platform Services firmware does not have any external interface, external commands are first received by the BMC over LAN and then relayed to the Platform Services firmware over IPMB channel. The BMC acts as a relay and the transport conversion device for these commands. For simplicity, the commands from the management console might be encapsulated in a generic CONFIG packet format (config data length, config data blob) to the BMC so that the BMC doesn't even have to parse the actual configuration data.

The BMC provides the access point for remote commands from external management SW and generates alerts to them. Intel® Intelligent Power Node Manager on Intel® Manageability Engine (Intel® ME) is an IPMI satellite controller. A mechanism needs to exist to forward commands to Intel® ME and send response back to originator. Similarly events from Intel® ME have to be sent as alerts outside of the BMC. It is the responsibility of BMC to implement these mechanisms for communication with Intel® Intelligent Power Node Manager.

The full specification can be downloaded from the following link:

<http://www.intel.com/content/dam/doc/technical-specification/intelligent-power-node-manager-1-5-specification.pdf>

## 2. Basic Decoding of a SEL Record

The System Event Log (SEL) record format is defined in the *IPMI Specification*. The following section provides a basic definition for each of the fields in a SEL. For more details see the *IPMI Specification*.

The definitions for the standard SEL can be found in Table 1.

The definitions for the OEM defined event logs can be found in Table 3 and Table 4.

### 2.1 Default Values in the SEL Records

Unless otherwise noted in the event record descriptions the following are the default values in all SEL entries.

- Byte [3] = Record Type (RT) = 02h = System event record
- Byte [9:8] = Generator ID = 0020h = BMC Firmware
- Byte [10] = Event Message Revision (ER) = 04h = IPMI 2.0

Table 1: SEL Record Format

Byte	Field	Description
1 2	Record ID (RID)	ID used for SEL Record access.
3	Record Type (RT)	[7:0] – Record Type 02h = System event record C0h-DFh = OEM timestamped, bytes 8-16 OEM defined (See Table 3) E0h-FFh = OEM non-timestamped, bytes 4-16 OEM defined (See Table 4)
4 5 6 7	Timestamp (TS)	Time when event was logged. LS byte first. Example: TS:[29][76][68][4C] = 4C687629h = 1281914409 = Sun, 15 Aug 2010 23:20:09 UTC <b>Note:</b> There are various websites that will convert the raw number to a date/time.

Byte	Field	Description
8 9	Generator ID ( <b>GID</b> )	<p>RqSA and LUN if event was generated from IPMB. Software ID if event was generated from system software.</p> <p><u>Byte 1</u> [7:1] – 7-bit I<sup>2</sup>C Slave Address, or 7-bit system software ID [0] 0b = ID is IPMB Slave Address 1b = System software ID Software ID values:</p> <ul style="list-style-type: none"> <li>▪ 0001h – BIOS POST for POST errors, RAS Configuration/State, Timestamp Synch, OS Boot events</li> <li>▪ 0033h – BIOS SMI Handler</li> <li>▪ 0020h – BMC Firmware</li> <li>▪ 002Ch – ME Firmware</li> <li>▪ 0041h – Server Management Software</li> <li>▪ 00C0h – HSC Firmware – HSBP A</li> <li>▪ 00C2h – HSC Firmware – HSBP B</li> </ul> <p><u>Byte 2</u> [7:4] – Channel number. Channel that event message was received over. 0h if the event message was received from the system interface, primary IPMB, or internally generated by the BMC. [3:2] – Reserved. Write as 00b. [1:0] – IPMB device LUN if byte 1 holds Slave Address. 00b otherwise.</p>
10	EvM Rev ( <b>ER</b> )	Event Message format version. 04h = IPMI v2.0; 03h = IPMI v1.0
11	Sensor Type ( <b>ST</b> )	Sensor Type Code for sensor that generated the event
12	Sensor # ( <b>SN</b> )	Number of sensor that generated the event (From SDR)
13	Event Dir   Event Type ( <b>EDIR</b> )	<p><u>Event Dir</u> [7] – 0b = Assertion event. 1b = Deassertion event.</p> <p><u>Event Type</u> Type of trigger for the event, for example, critical threshold going high, state asserted, and so on. Also indicates class of the event. For example, discrete, threshold, or OEM. The Event Type field is encoded using the Event/Reading Type Code.</p> <p>[6:0] – Event Type Codes 01h = Threshold (States = 0x00-0x0b) 02h-0ch = Discrete 6Fh = Sensor-Specific 70-7Fh = OEM</p>
14	Event Data 1 ( <b>ED1</b> )	Per Table 2: Event Request Message Event Data Field Contents
15	Event Data 2 ( <b>ED2</b> )	
16	Event Data 3 ( <b>ED3</b> )	

Table 2: Event Request Message Event Data Field Contents

Sensor Class	Event Data
Threshold	<p><u>Event Data 1</u></p> <p>[7:6] – 00b = Unspecified Event Data 2  01b = Trigger reading in Event Data 2  10b = OEM code in Event Data 2  11b = Sensor-specific event extension code in Event Data 2</p> <p>[5:4] – 00b = Unspecified Event Data 3  01b = Trigger threshold value in Event Data 3  10b = OEM code in Event Data 3  11b = Sensor-specific event extension code in Event Data 3</p> <p>[3:0] – Offset from Event/Reading Code for threshold event.</p> <p><u>Event Data 2</u> – Reading that triggered event, FFh or not present if unspecified.</p> <p><u>Event Data 3</u> – Threshold value that triggered event, FFh or not present if unspecified. If present, Event Data 2 must be present.</p>
discrete	<p><u>Event Data 1</u></p> <p>[7:6] – 00b = Unspecified Event Data 2  01b = Previous state and/or severity in Event Data 2  10b = OEM code in Event Data 2  11b = Sensor-specific event extension code in Event Data 2</p> <p>[5:4] – 00b = Unspecified Event Data 3  01b = Reserved  10b = OEM code in Event Data 3  11b = Sensor-specific event extension code in Event Data 3</p> <p>[3:0] – Offset from Event/Reading Code for discrete event state</p> <p><u>Event Data 2</u></p> <p>[7:4] – Optional offset from “Severity” Event/Reading Code (0Fh if unspecified).</p> <p>[3:0] – Optional offset from Event/Reading Type Code for previous discrete event state (0Fh if unspecified).</p> <p><u>Event Data 3</u> – Optional OEM code. FFh or not present if unspecified.</p>
OEM	<p><u>Event Data 1</u></p> <p>[7:6] – 00b = Unspecified in Event Data 2  01b = Previous state and/or severity in Event Data 2  10b = OEM code in Event Data 2  11b = Reserved</p> <p>[5:4] – 00b = Unspecified Event Data 3  01b = Reserved  10b = OEM code in Event Data 3  11b = Reserved</p> <p>[3:0] – Offset from Event/Reading Type Code</p> <p><u>Event Data 2</u></p> <p>[7:4] – Optional OEM code bits or offset from “Severity” Event/Reading Type Code (0Fh if unspecified).</p> <p>[3:0] – Optional OEM code or offset from Event/Reading Type Code for previous event state (0Fh if unspecified).</p> <p><u>Event Data 3</u> – Optional OEM code. FFh or not present or unspecified.</p>



Table 3: OEM SEL Record (Type C0h-DFh)

Byte	Field	Description
1 2	Record ID ( <b>RID</b> )	ID used for SEL Record access.
3	Record Type ( <b>RT</b> )	[7:0] – Record Type C0h-DFh = OEM timestamped, bytes 8-16 OEM defined
4 5 6 7	Timestamp ( <b>TS</b> )	Time when event was logged. LS byte first. Example: TS:[29][76][68][4C] = 4C687629h = 1281914409 = Sun, 15 Aug 2010 23:20:09 UTC <b>Note:</b> There are various websites that will convert the raw number to a date/time.
8 9 10	Manufacturer ID	LS Byte first. The manufacturer ID is a 20-bit value that is derived from the IANA “Private Enterprise” ID. Most significant four bits = Reserved (0000b). 000000h = Unspecified. 0FFFFFFh = Reserved. This value is binary encoded. For example the ID for the IPMI forum is 7154 decimal, which is 1BF2h, which will be stored in this record as F2h, 1Bh, 00h for bytes 8 through 10, respectively.
11 12 13 14 15 16	OEM Defined	OEM Defined. This is defined according to the manufacturer identified by the Manufacturer ID field.

Table 4: OEM SEL Record (Type E0h-FFh)

Byte	Field	Description
1 2	Record ID ( <b>RID</b> )	ID used for SEL Record access.
3	Record Type ( <b>RT</b> )	[7:0] – Record Type E0h-FFh = OEM system event record
4 5 6 7 8 9 10 11 12 13 14 15 16	OEM	OEM Defined. This is defined by the system integrator.

### 3. Sensor Cross Reference List

This section contains a cross reference to help find details on any specific SEL entry.

#### 3.1 BMC owned Sensors (GID = 0020h)

The following table can be used to find the details of sensors owned by the BMC.

Table 5: BMC owned Sensors

Sensor Number	Sensor Name	Details Section	Next Steps
01h	Power Unit Status ( <i>Pwr Unit Status</i> )	<a href="#">Power Unit Status Sensor</a>	<a href="#">Table 16: Power Unit Status Sensor – Sensor Specific Offsets – Next Steps</a>
02h	Power Unit Redundancy ( <i>Pwr Unit Redund</i> )	<a href="#">Power Unit Redundancy Sensor</a>	<a href="#">Table 18: Power Unit Redundancy Sensor – Event Trigger Offset – Next Steps</a>
03h	IPMI Watchdog ( <i>IPMI Watchdog</i> )	<a href="#">IPMI Watchdog</a>	<a href="#">Table 77: IPMI Watchdog Sensor Event Trigger Offset – Next Steps</a>
04h	Physical Security ( <i>Physical Scrty</i> )	<a href="#">Physical Security</a>	<a href="#">Table 73: Physical Security Sensor Event Trigger Offset – Next Steps</a>
05h	FP Interrupt ( <i>FP NMI Diag Int</i> )	<a href="#">FP (NMI) Interrupt</a>	<a href="#">FP (NMI) Interrupt – Next Steps</a>
06h	SMI Timeout ( <i>SMI Timeout</i> )	<a href="#">SMI Timeout</a>	<a href="#">SMI Timeout – Next Steps</a>
07h	System Event Log ( <i>System Event Log</i> )	<a href="#">System Event Log Cleared</a>	Not applicable
08h	System Event ( <i>System Event</i> )	<a href="#">System Event – PEF Action</a>	<a href="#">System Event – PEF Action – Next Steps</a>
09h	Button Press Event ( <i>Button Press</i> )	<a href="#">Button Press Events</a>	Not applicable

Sensor Number	Sensor Name	Details Section	Next Steps
10h	BB +1.1V IOH (BB +1.1V IOH)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
11h	BB +1.1V P1 Vccp (BB +1.1V P1 Vccp)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
12h	BB +1.1V P2 Vccp (BB +1.1V P2 Vccp)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
13h	BB +1.5V P1 DDR3 (BB +1.5V P1 DDR3)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
14h	BB +1.5V P2 DDR3 (BB +1.5V P2 DDR3)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
15h	BB +1.8V AUX (BB +1.8V AUX)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
16h	BB +3.3V (BB +3.3V)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
17h	BB +3.3V STBY (BB +3.3V STBY)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
18h	BB +3.3V Vbat (BB +3.3V Vbat)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
19h	BB +5.0V (BB +5.0V)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
1Ah	BB +5.0V STBY (BB +5.0V STBY)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
1Bh	BB +12.0V (BB +12.0V)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
1Ch	BB -12.0V (BB -12.0V)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
1Dh	BB +1.35V P1 LV DDR3 (BB +1.35v P1 MEM)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>

Sensor Number	Sensor Name	Details Section	Next Steps
1Eh	BB +1.35V P2 LV DDR3 (BB +1.35v P2 MEM)	<a href="#">Voltage Sensors</a>	<a href="#">Table 14: Voltage Sensors – Next Steps</a>
20h	Baseboard Temperature (Baseboard Temp)	<a href="#">Regular Temperature Sensors</a>	<a href="#">Table 35: Temperature Sensors – Next Steps</a>
21h	Front Panel Temperature (Front Panel Temp)	<a href="#">Regular Temperature Sensors</a>	<a href="#">Table 35: Temperature Sensors – Next Steps</a>
22h	IOH Thermal Margin (IOH Therm Margin)	<a href="#">Thermal Margin Sensors</a>	<a href="#">Table 38: Thermal Margin Sensors – Next Steps</a>
23h	Processor 1 Memory Thermal Margin (Mem P1 Thrm Mrgn)	<a href="#">Thermal Margin Sensors</a>	<a href="#">Table 38: Thermal Margin Sensors – Next Steps</a>
24h	Processor 2 Memory Thermal Margin (Mem P2 Thrm Mrgn)	<a href="#">Thermal Margin Sensors</a>	<a href="#">Table 38: Thermal Margin Sensors – Next Steps</a>
30h-39h	Fan Tachometer Sensors (Chassis specific sensor names)	<a href="#">Fan Speed Sensors</a>	<a href="#">Table 28: Fan Speed Sensor – Event Trigger Offset – Next Steps</a>
40h-45h	Fan Present Sensors (Fan x Present)	<a href="#">Fan Presence and Redundancy Sensors</a>	<a href="#">Table 30: Fan Presence Sensors – Event Trigger Offset – Next Steps</a>
46h	Fan Redundancy (Fan Redundancy)	<a href="#">Fan Presence and Redundancy Sensors</a>	<a href="#">Table 32: Fan Redundancy Sensor – Event Trigger Offset – Next Steps</a>
50h	Power Supply 1 Status (PS1 Status)	<a href="#">Power Supply Status Sensors</a>	<a href="#">Table 16: Power Unit Status Sensor – Sensor Specific Offsets – Next Steps</a>
51h	Power Supply 2 Status (PS2 Status)	<a href="#">Power Supply Status Sensors</a>	<a href="#">Table 16: Power Unit Status Sensor – Sensor Specific Offsets – Next Steps</a>
52h	Power Supply 1 AC Power Input (PS1 Power In)	<a href="#">Power Supply AC Power Input Sensors</a>	<a href="#">Table 22: Power Supply AC Power Input Sensor – Event Trigger Offset – Next Steps</a>
53h	Power Supply 2 AC Power Input (PS2 Power In)	<a href="#">Power Supply AC Power Input Sensors</a>	<a href="#">Table 22: Power Supply AC Power Input Sensor – Event Trigger Offset – Next Steps</a>

Sensor Number	Sensor Name	Details Section	Next Steps
54h	Power Supply 1 +12V % of Maximum Current Output (PS1 Curr Out %)	<a href="#">Power Supply Current Output % Sensors</a>	<a href="#">Table 24: Power Supply Current Output % Sensor – Event Trigger Offset – Next Steps</a>
55h	Power Supply 2 +12V % of Maximum Current Output (PS2 Curr Out %)	<a href="#">Power Supply Current Output % Sensors</a>	<a href="#">Table 24: Power Supply Current Output % Sensor – Event Trigger Offset – Next Steps</a>
56h	Power Supply 1 Temperature (PS1 Temperature)	<a href="#">Power Supply Temperature Sensors</a>	<a href="#">Table 26: Power Supply Temperature Sensor – Event Trigger Offset – Next Steps</a>
57h	Power Supply 2 Temperature (PS2 Temperature)	<a href="#">Power Supply Temperature Sensors</a>	<a href="#">Table 26: Power Supply Temperature Sensor – Event Trigger Offset – Next Steps</a>
60h	Processor 1 Status (P1 Status)	<a href="#">Processor Status Sensor</a>	<a href="#">Table 45: Processor Status Sensors – Next Steps</a>
61h	Processor 2 Status (P2 Status)	<a href="#">Processor Status Sensor</a>	<a href="#">Table 45: Processor Status Sensors – Next Steps</a>
62h	Processor 1 Thermal Margin (P1 Therm Margin)	<a href="#">Thermal Margin Sensors</a>	<a href="#">Table 38: Thermal Margin Sensors – Next Steps</a>
63h	Processor 2 Thermal Margin (P2 Therm Margin)	<a href="#">Thermal Margin Sensors</a>	<a href="#">Table 38: Thermal Margin Sensors – Next Steps</a>
64h	Processor 1 Thermal Control % (P1 Therm Ctrl %)	<a href="#">Processor Thermal Control % Sensors</a>	<a href="#">Table 41: Processor Thermal Control % Sensors – Next Steps</a>
65h	Processor 2 Thermal Control % (P2 Therm Ctrl %)	<a href="#">Processor Thermal Control % Sensors</a>	<a href="#">Table 41: Processor Thermal Control % Sensors – Next Steps</a>
66h	Processor 1 VRD Temp (P1 VRD Hot)	<a href="#">Discrete Thermal Sensors</a>	<a href="#">Table 43: Discrete Thermal Sensors</a>
67h	Processor 2 VRD Temp (P2 VRD Hot)	<a href="#">Discrete Thermal Sensors</a>	<a href="#">Table 43: Discrete Thermal Sensors</a>
68h	Catastrophic Error (CATERR)	<a href="#">Catastrophic Error Sensor</a>	<a href="#">Catastrophic Error Sensor– Next Steps</a>
69h	CPU Missing (CPU Missing)	<a href="#">CPU Missing Sensor</a>	<a href="#">CPU Missing Sensor – Next Steps</a>

Sensor Number	Sensor Name	Details Section	Next Steps
6Ah	IOH Thermal Trip (IOH Thermal Trip)	<a href="#">Discrete Thermal Sensors</a>	<a href="#">Table 43: Discrete Thermal Sensors</a>

### 3.2 BIOS POST owned Sensors (GID = 0001h)

The following table can be used to find the details of sensors owned by BIOS POST.

Table 6: BIOS POST owned Sensors

Sensor Number	Sensor Name	Details Section	Next Steps
01h	Mirroring Redundancy State	<a href="#">Mirrored Redundancy State Sensor</a>	<a href="#">Table 55: Mirrored Redundancy State Sensor Event Trigger Offset – Next Steps</a>
06h	POST Error	<a href="#">System Firmware Progress (Formerly Post Error)</a>	<a href="#">System Firmware Progress (Formerly Post Error) – Next Steps</a>
11h	Sparing Redundancy State	<a href="#">Sparing Redundancy State Sensor</a>	<a href="#">Table 59: Sparing Redundancy State Sensor Event Trigger Offset – Next Steps</a>
12h	Mirroring Configuration Status	<a href="#">Mirroring Configuration Status</a>	<a href="#">Table 53: Mirroring Configuration Status Sensor Event Trigger Offset – Next Steps</a>
13h	Sparing Configuration Status	<a href="#">Sparing Configuration Status</a>	<a href="#">Table 57: Sparing Configuration Status Sensor Event Trigger Offset – Next Steps</a>
83h	System Event	<a href="#">System Events</a>	Not applicable

### 3.3 BIOS SMI owned Sensors (GID = 0033h)

The following table can be used to find the details of sensors owned by BIOS SMI.

Table 7: BIOS SMI owned Sensors

Sensor Number	Sensor Name	Details Section	Next Steps
02h	Memory ECC Error	<a href="#">Memory Correctable and Uncorrectable ECC Error</a>	<a href="#">Table 61: Correctable and Uncorrectable ECC Error Sensor Event Trigger Offset – Next Steps</a>
03h	Legacy PCI Error	<a href="#">Legacy PCI Errors</a>	<a href="#">Table 68: Legacy PCI Error Sensor Event Trigger Offset – Next Steps</a>
04h	PCI Express Fatal Error	<a href="#">PCI Express Fatal Errors</a>	<a href="#">Table 66: PCI Express* Fatal Error Sensor Event Trigger Offset – Next Steps</a>
05h	PCI Express Correctable Error	<a href="#">PCI Express Correctable errors</a>	<a href="#">Table 64: PCI Express* Correctable Error Sensor Event Trigger Offset – Next Steps</a>
06h	Intel® QuickPath Interface Correctable Error	<a href="#">QPI Correctable Error Sensor</a>	<a href="#">QPI Correctable Error Sensor – Next Steps</a>
07h	Intel® QuickPath Interface Non-fatal Error	<a href="#">QPI Non-Fatal Error Sensor</a>	<a href="#">QPI Non-Fatal Error Sensor – Next Steps</a>
14h	Memory Address Parity Error	<a href="#">Memory Address Parity Error</a>	<a href="#">Memory Address Parity Error Sensor Next Steps</a>
17h	Intel® QuickPath Interface Fatal Error	<a href="#">QPI Fatal and Fatal #2</a>	<a href="#">QPI Fatal and Fatal #2 – Next Steps</a>
18h	Intel® QuickPath Interface Fatal2 Error	<a href="#">QPI Fatal and Fatal #2</a>	<a href="#">QPI Fatal and Fatal #2 – Next Steps</a>
83h	System Event	<a href="#">System Events</a>	Not applicable

### 3.4 Hot Swap Controller Firmware owned Sensors (GID = 00C0h/00C2h)

The following table can be used to find the details of sensors owned by the Hot Swap Controller (HSC) firmware. The HSC firmware resides on a Hot Swap Back Plane (HSBP). There can be up to two HSBPs in a system. Each HSBP will have its own GID.

- 00C0h = HSC Firmware – HSBP A
- 00C2h = HSC Firmware – HSBP B

Table 8: Hot Swap Controller Firmware owned Sensors

Sensor Number	Sensor Name	Details Section	Next Steps
01h	Backplane Temperature	<a href="#">HSC Backplane Temperature Sensor</a>	<a href="#">Table 82: HSC Backplane Temperature Sensor – Event Trigger Offset – Next Steps</a>
02h	Drive Slot 0 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
03h	Drive Slot 1 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
04h	Drive Slot 2 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
05h	Drive Slot 3 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
06h	Drive Slot 4 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
07h	Drive Slot 5 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
6 Slot HSBP			
08h	Drive Slot 0 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
09h	Drive Slot 1 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Ah	Drive Slot 2 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Bh	Drive Slot 3 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Ch	Drive Slot 4 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Dh	Drive Slot 5 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
8 Slot HSBP			
08h	Drive Slot 6 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>



Sensor Number	Sensor Name	Details Section	Next Steps
09h	Drive Slot 7 Status	<a href="#">HSC Drive Slot Status Sensor</a>	<a href="#">HSC Drive Slot Status Sensor – Next Steps</a>
0Ah	Drive Slot 0 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Bh	Drive Slot 1 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Ch	Drive Slot 2 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Dh	Drive Slot 3 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Eh	Drive Slot 4 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
0Fh	Drive Slot 5 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
10h	Drive Slot 6 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>
11h	Drive Slot 7 Presence	<a href="#">HSC Drive Presence Sensor</a>	<a href="#">HSC Drive Presence Sensor – Next Steps</a>

### 3.5 Node Manager / ME Firmware owned Sensors (GID = 002Ch or 602Ch)

The following table can be used to find the details of sensors owned by the Node Manager / Management Engine (ME) firmware.

Table 9: Management Engine Firmware owned Sensors

Sensor Number	Sensor Name	Details Section	Next Steps
17h	ME Firmware Health Events	<a href="#">ME Firmware Health Event</a>	<a href="#">ME Firmware Health Event – Next Steps</a>
18h	Node Manager Exception Events	<a href="#">Node Manager Exception Event</a>	<a href="#">Node Manager Exception Event – Next Steps</a>
19h	Node Manager Health Events	<a href="#">Node Manager Health Event</a>	<a href="#">Node Manager Health Event – Next Steps</a>
1Ah	Node Manager Operational Capabilities Change Events	<a href="#">Node Manager Operational Capabilities Change</a>	<a href="#">Node Manager Operational Capabilities Change – Next Steps</a>
1Bh	Node Manager Alert Threshold Exceeded Events	<a href="#">Node Manager Alert Threshold Exceeded</a>	<a href="#">Node Manager Alert Threshold Exceeded – Next Steps</a>

### 3.6 Microsoft\* OS owned Events (GID = 0041)

The following table can be used to find the details of records that are owned by the Microsoft\* Operating System (OS).

Table 10: Microsoft\* OS owned Events

Sensor Name	Record Type	Sensor Type	Details Section	Next Steps
Boot Event	02h	1Fh = OS Boot	<a href="#">Table 91: Boot-up Event Record Typical Characteristics</a>	Not applicable
	DCh	Not applicable	<a href="#">Table 92: Boot-up OEM Event Record Typical Characteristics</a>	
Shutdown Event	02h	20h = OS Stop/Shutdown	<a href="#">Table 93: Shutdown Reason Code Event Record Typical Characteristics</a>	Not applicable
	DDh	Not applicable	<a href="#">Table 94: Shutdown Reason OEM Event Record Typical Characteristics</a> <a href="#">Table 95: Shutdown Comment OEM Event Record Typical Characteristics</a>	Not applicable
Bug Check / Blue Screen	02h	20h = OS Stop/Shutdown	<a href="#">Table 96: Bug Check / Blue Screen – OS Stop Event Record Typical Characteristics</a>	Not applicable
	DEh	Not applicable	<a href="#">Table 97: Bug Check / Blue Screen Code OEM Event Record Typical Characteristics</a>	

### 3.7 Linux\* Kernel Panic Events (GID = 0021)

The following table can be used to find the details of records that can be generated when there is a Linux\* Kernel panic.

Table 11: Linux\* Kernel Panic Events

Sensor Name	Record Type	Sensor Type	Details Section	Next Steps
Linux* Kernel Panic	02h	20h = OS Stop/Shutdown	<a href="#">Table 98: Linux* Kernel Panic Event Record Characteristics</a>	Not applicable
	F0h	Not applicable	<a href="#">Table 99: Linux* Kernel Panic String Extended Record Characteristics</a>	

## 4. Power Subsystems

The BMC monitors the power subsystem including power supplies, select onboard voltages, and related sensors.

### 4.1 Voltage Sensors

The BMC monitors the main voltage sources in the system, including the baseboard, memory, and processors, using IPMI-compliant analog/threshold sensors.

**Note:** A voltage error could be caused by the device supplying the voltage or by the device using the voltage. For each sensor it will be noted who is supplying the voltage and who is using it.

Table 12: Voltage Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	02h = Voltage
12	Sensor Number	See Table 14
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Triggers as described in Table 13
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 13: Voltage Sensors Event Triggers – Description

Event Trigger		Assertion Severity	Deassert Severity	Description
Hex	Description			
00h	Lower non-critical going low	Degraded	OK	The voltage has dropped below its lower non-critical threshold.
02h	Lower critical going low	non-fatal	Degraded	The voltage has dropped below its lower critical threshold.
07h	Upper non-critical going high	Degraded	OK	The voltage has gone over its upper non-critical threshold.
09h	Upper critical going high	non-fatal	Degraded	The voltage has gone over its upper critical threshold.

Table 14: Voltage Sensors – Next Steps

Sensor Number	Sensor Name	Next Steps
10h	BB +1.1V IOH	<p>This 1.1V line is supplied by the main board. This 1.1V line is used by the I/O hub (IOH)</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. If the issue remains, replace the motherboard.</li> </ol>
11h	BB +1.1V P1 Vccp	<p>This 1.1V line is supplied by the main board. This 1.1V line is used by processor 1.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Cross test the processor if possible. If the issue remains with the socket, replace the main board, otherwise the processor.</li> </ol>
12h	BB +1.1V P2 Vccp	<p>This 1.1V line is supplied by the main board. This 1.1V line is used by processor 2.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Cross test the processor if possible. If the issue remains with the socket, replace the main board, otherwise the processor.</li> </ol>

Sensor Number	Sensor Name	Next Steps
13h	BB +1.5V P1 DDR3	<p>This 1.5V line is supplied by the main board. This 1.5V line is used by the memory on processor 1.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Check the DIMMs are seated properly.</li> <li>3. Cross test the DIMMs. If the issue remains with the DIMMs on this socket, replace the main board, otherwise replace the DIMM.</li> </ol>
14h	BB +1.5V P2 DDR3	<p>This 1.5V line is supplied by the main board. This 1.5V line is used by the memory on processor 2.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Check the DIMMs are seated properly.</li> <li>3. Cross test the DIMMs. If the issue remains with the DIMMs on this socket, replace the main board, otherwise replace the DIMM.</li> </ol>
15h	BB +1.8V AUX	<p>+1.8V is supplied by the main board. +1.8V is used by the onboard NIC and I/O hub.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. If the issue remains, replace the main board.</li> </ol>
16h	BB +3.3V	<p>+3.3V is supplied by the power supplies. +3.3V is used by the PCIe and PCI-X slots.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Reseat any PCI cards, and try them in other slots.</li> <li>3. If the issue follows the card, swap it, otherwise, replace the main board.</li> <li>4. If the issue remains, replace the power supplies.</li> </ol>
17h	BB +3.3V STBY	<p>+3.3V Stby is supplied by the main board. +3.3V Stby is used by the BMC, Onboard NIC, IOH, and ICH.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. If the issue remains, replace the board.</li> <li>3. If the issue remains, replace the power supplies.</li> </ol>

Sensor Number	Sensor Name	Next Steps
18h	BB +3.3V Vbat	<p>+3.3V Vbat is supplied by the CMOS battery when power is off and by the main board when power is on.</p> <p>+3.3V Vbat is used by the CMOS and related circuits.</p> <ol style="list-style-type: none"> <li>1. Replace the CMOS battery. Any battery of type CR2032 can be used.</li> <li>2. If error remains (unlikely), replace the board.</li> </ol>
19h	BB +5.0V	<p>+5.0V is supplied by the power supplies.</p> <p>+5.0V is used by the PCI slots.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Reseat any PCI cards, and try them in other slots.</li> <li>3. If the issue follows the card, swap it, otherwise, replace the main board.</li> <li>4. If the issue remains, replace the power supplies.</li> </ol>
1Ah	BB +5.0V STBY	<p>+5.0V STBY is supplied by the power supplies.</p> <p>+5.0V STBY is used to generate other standby voltages.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. If the issue remains, replace the board.</li> <li>3. If the issue remains, replace the power supplies.</li> </ol>
1Bh	BB +12.0V	<p>+12V is supplied by the power supplies.</p> <p>+12V is used by SATA drives, Fans, and PCI cards. In addition it is used to generate various processor voltages.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Check connections on fans and HDDs.</li> <li>3. If the issue follows the component, swap it, otherwise, replace the board.</li> <li>4. If the issue remains, replace the power supplies.</li> </ol>
1Ch	BB -12.0V	<p>-12V is supplied by the power supplies.</p> <p>-12V is used by the serial port and by PCI cards. In addition it is used to generate various processor voltages.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Reseat any PCI cards, and try them in other slots.</li> <li>3. If the issue follows the card, swap it, otherwise, replace the main board.</li> <li>4. If the issue remains, replace the power supplies.</li> </ol>

Sensor Number	Sensor Name	Next Steps
1Dh	BB +1.35 P1 Mem	<p>This 1.35V line is supplied by the main board. This 1.35V line is used by low voltage memory on processor 1.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Check the DIMMs are seated properly.</li> <li>3. Cross test the DIMMs.</li> <li>4. If the issue remains with the DIMMs on this socket, replace the main board, otherwise the DIMM.</li> </ol>
1Eh	BB +1.35 P2 Mem	<p>This 1.35V line is supplied by the main board. This 1.35V line is used by low voltage memory on processor 2.</p> <ol style="list-style-type: none"> <li>1. Ensure all cables are connected correctly.</li> <li>2. Check the DIMMs are seated properly.</li> <li>3. Cross test the DIMMs.</li> <li>4. If the issue remains with the DIMMs on this socket, replace the main board, otherwise the DIMM.</li> </ol>

## 4.2 Power Unit

The power unit monitors the power state of the system and logs the state changes in the SEL.

### 4.2.1 Power Unit Status Sensor

The power unit status sensor monitors the power state of the system and logs state changes. Expected power-on events such as DC ON/OFF are logged and unexpected events are also logged, such as AC loss and power good loss.

Table 15: Power Unit Status Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	09h = Power Unit
12	Sensor Number	01h

Byte	Field	Description
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] = Sensor Specific offset as described in Table 9
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 16: Power Unit Status Sensor – Sensor Specific Offsets – Next Steps

Sensor Specific Offset		Description	Next Steps
Hex	Description		
00h	Power down	System is powered down.	Informational Event
04h	AC Lost	AC removed.	Informational Event
05h	Soft Power Control Failure	Generally means power good was lost in the system, causing a shutdown.	This could be caused by the power supply subsystem or system components. <ol style="list-style-type: none"> <li>1. Verify all power cables and adapters are connected properly (AC cables as well as the cables between the PSU and system components).</li> <li>2. Cross test the PSU if possible.</li> <li>3. Replace the power subsystem.</li> </ol>
06h	Power Unit Failure	Power subsystem experienced a failure.	Indicates a power supply failed. <ol style="list-style-type: none"> <li>1. Remove and reapply AC power.</li> <li>2. If the power supply still fails, replace it.</li> </ol>

## 4.2.2 Power Unit Redundancy Sensor

This sensor is enabled on systems that support redundant power supplies. When a system has AC applied or if it loses redundancy of the power supplies a message will get logged into the SEL.



Table 17: Power Unit Redundancy Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	09h = Power Unit
12	Sensor Number	02h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 0Bh (Generic Discrete)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 18
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 18: Power Unit Redundancy Sensor – Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Fully redundant	System is fully operational.	Informational Event
01h	Redundancy lost	System is not running in redundant power supply mode.	This event should be accompanied by specific power supply errors (AC lost, PSU failure, and so on). Troubleshoot these events accordingly.
02h	Redundancy degraded		
03h	Non-redundant, sufficient from redundant		
04h	Non-redundant, sufficient from insufficient		
05h	Non-redundant, insufficient		
06h	Non-redundant, degraded from fully redundant		
07h	Redundant, degraded from non-redundant		

## 4.3 Power Supply

The BMC monitors the power supply subsystem.

### 4.3.1 Power Supply Status Sensors

These sensors report the status of the power supplies in the system. When a system first AC applied or removed it can log an event. Also if there is a failure, predictive failure, or a configuration error it can log an event.

Table 19: Power Supply Status Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	08h = Power Supply
12	Sensor Number	50h = Power Supply 1 Status 51h = Power Supply 2 Status
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] = Sensor Specific offset as described in Table 20
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 20: Power Supply Status Sensor – Sensor Specific Offsets – Next Steps

Sensor Specific Offset		Description	Next Steps
Hex	Description		
00h	Presence	Power supply detected.	Informational Event.

Sensor Specific Offset		Description	Next Steps
Hex	Description		
01h	Failure	Power supply failed.	Indicates a power supply failed. 1) Remove and reapply AC. 2) If the power supply still fails, replace it.
02h	Predictive Failure	Typically means a fan inside the power supply is not cooling the power supply. It may indicate the fan is failing.	Replace the power supply.
03h	AC lost	AC removed.	Informational Event.
06h	Configuration error	Power supply configuration is not supported.	Indicates that at least one of the supplies is not correct for your system configuration. 1) Remove the power supply and verify compatibility. 2) If the power supply is compatible it may be faulty. Replace it.

### 4.3.2 Power Supply AC Power Input Sensors

These sensors will log an event when a power supply in the system is exceeding its AC power in threshold.

Table 21: Power Supply AC Power Input Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	0Bh = Other Units
12	Sensor Number	52h = Power Supply 1 AC Power Input 53h = Power Supply 2 AC Power Input
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Trigger Offset as described in Table 22

Byte	Field	Description
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 22: Power Supply AC Power Input Sensor – Event Trigger Offset – Next Steps

Event Trigger Offset		Assertion Severity	Deassert Severity	Description	Next Steps
Hex	Description				
07h	Upper non-critical going high	Degraded	OK	PMBus* feature to monitor power supply power consumption.	If you see this event, the system is pulling too much power on the input for the PSU rating. 1. Verify the power budget is within the specified range. 2. Check <a href="http://www.intel.com/p/en_US/support/">http://www.intel.com/p/en_US/support/</a> for the power budget tool for your system.
09h	Upper critical going high	non-fatal	Degraded		

### 4.3.3 Power Supply Current Output % Sensors

PMBus\*-compliant power supplies may monitor the current output of the main 12v voltage rail and report the current usage as a percentage of the maximum power output for that rail.

Table 23: Power Supply Current Output % Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	03h = Current
12	Sensor Number	54h = Power Supply 1 Current Output % 55h = Power Supply 2 Current Output %

Byte	Field	Description
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Trigger Offset as described in Table 24
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 24: Power Supply Current Output % Sensor – Event Trigger Offset – Next Steps

Event Trigger Offset		Assertion Severity	Deassert Severity	Description	Next Steps
Hex	Description				
07h	Upper non-critical going high	Degraded	OK	PMBus* feature to monitor power supply power consumption.	If you see this event, the system is using too much power on the output for the PSU rating. <ol style="list-style-type: none"> <li>1. Verify the power budget is within the specified range.</li> <li>2. Check <a href="http://www.intel.com/p/en_US/support/">http://www.intel.com/p/en_US/support/</a> for the power budget tool for your system.</li> </ol>
09h	Upper critical going high	non-fatal	Degraded		

#### 4.3.4 Power Supply Temperature Sensors

The BMC monitors one power supply temperature sensor for each installed PMBus\*-compliant power supply.

Table 25: Power Supply Temperature Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	01h = Temperature
12	Sensor Number	56h = Power Supply 1 Temperature 57h = Power Supply 2 Temperature
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Trigger Offset as described in Table 26
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 26: Power Supply Temperature Sensor – Event Trigger Offset – Next Steps

Event Trigger Offset		Assertion Severity	Deassert Severity	Description	Next Steps
Hex	Description				
07h	Upper non-critical going high	Degraded	OK	An upper non-critical or critical temperature threshold has been crossed.	<ol style="list-style-type: none"> <li>1. Check for clear and unobstructed airflow into and out of the chassis.</li> <li>2. Ensure the SDR is programmed and correct chassis has been selected.</li> <li>3. Ensure there are no fan failures.</li> <li>4. Ensure the air used to cool the system is within the thermal specifications for the system (typically below 35°C).</li> </ol>
09h	Upper critical going high	non-fatal	Degraded		

## 5. Cooling Subsystem

### 5.1 Fan Sensors

There are three types of fan sensors that can be present on Intel® Server Systems: speed, presence, and redundancy. The last two are only present in systems with hot-swap redundant fans.

#### 5.1.1 Fan Speed Sensors

Fan speed sensors monitor the rpm signal on the relevant fan headers on the platform. Fan speed sensors are threshold-based sensors. Usually they only have lower (critical) thresholds set, so that a SEL entry is only generated if the fan spins too slowly.

Table 27: Fan Speed Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	04h = Fan
12	Sensor Number	30h-39h (Chassis specific)
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Trigger Offset as described in Table 28
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 28: Fan Speed Sensor – Event Trigger Offset – Next Steps

Event Trigger Offset		Assertion Severity	Deassert Severity	Description	Next Steps
Hex	Description				
00h	Lower non-critical going low	Degraded	OK	The fan speed has dropped below its lower non-critical threshold.	<p>A fan speed error on a new system build is typically not caused by the fan spinning too slowly, instead it is caused by the fan being connected to the wrong header (the BMC expects them on certain headers for each chassis and will log this event if there is no fan on that header).</p> <ol style="list-style-type: none"> <li>1. Refer to the <i>Quick Start Guide</i> or the <i>Service Guide</i> to identify the correct fan headers to use.</li> <li>2. Ensure the latest FRUSDR update has been run and the correct chassis was detected or selected.</li> <li>3. If you are sure this was done, the event may be a sign of impending fan failure (although this will only normally apply if the system has been in use for a while). Replace the fan.</li> </ol>
02h	Lower critical going low	non-fatal	Degraded	The fan speed has dropped below its lower critical threshold.	

### 5.1.2 Fan Presence and Redundancy Sensors

Fan presence sensors are only implemented for hot-swap fans, and require an additional pin on the fan header. Fan redundancy is an aggregate of the fan presence sensors and will warn when redundancy is lost. Typically the redundancy mode on Intel® servers is an n+1 redundancy (if one fan fails there are still sufficient fans to cool the system, but it is no longer redundant) although other modes are also possible.

Table 29: Fan Presence Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	04h = Fan
12	Sensor Number	40h-45h (Chassis specific)
13	Event Direction and Event Type	<p>[7] Event direction            0b = Assertion Event            1b = Deassertion Event</p> <p>[6:0] Event Type = 08h (Generic “digital” Discrete)</p>



Byte	Field	Description
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 30
15	Event Data 2	Not used
16	Event Data 3	Not used

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 30: Fan Presence Sensors – Event Trigger Offset – Next Steps

Event Trigger Offset		Assertion Severity	Deassert Severity	Description	Next Steps
Hex	Description				
01h	Device Present	OK	Degraded	Assertion – A fan was inserted. This event may also get logged when the BMC initializes when AC is applied.	Informational only
				Deassert – A fan was removed, or was not present at the expected location when the BMC initialized.	These events only get generated in systems with hot-swappable fans, and normally only when a fan is physically inserted or removed. If fans were not physically removed: <ol style="list-style-type: none"> <li>1. Use the <i>Quick Start Guide</i> to check whether the right fan headers were used.</li> <li>2. Swap the fans round to see whether the problem stays with the location, or follows the fan.</li> <li>3. Replace the fan or fan wiring/housing depending on the outcome of step 2.</li> <li>4. Ensure the latest FRUSDR update has been run and the correct chassis was detected or selected.</li> </ol>

Table 31: Fan Redundancy Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	04h = Fan
12	Sensor Number	46h

Byte	Field	Description
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 0Bh (Generic Discrete)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 32
15	Event Data 2	Not used
16	Event Data 3	Not used

The following table describes the severity of each of the event triggers for both assertion and deassertion.

Table 32: Fan Redundancy Sensor – Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Fully redundant	System has lost one or more fans and is running in non-redundant mode. There are enough fans to keep the system properly cooled, but fan speeds will boost.	Fan redundancy loss indicates failure of one or more fans.  Look for lower (non) critical fan errors, or fan removal errors in the SEL, to indicate which fan is causing the problem, and follow the troubleshooting steps for these event types.
01h	Redundancy lost		
02h	Redundancy degraded		
03h	Non-redundant, sufficient from redundant		
04h	Non-redundant, sufficient from insufficient		
05h	Non-redundant, insufficient	System has lost fans and may no longer be able to cool itself adequately. Overheating may occur if this situation remains for a longer period of time.	
06h	Non-redundant, degraded from fully redundant	System has lost one or more fans and is running in non-redundant mode. There are enough fans to keep the system properly cooled, but fan speeds will boost.	
07h	Redundant, degraded from non-redundant	System has lost one or more fans and is running in a degraded mode, but still is redundant. There are enough fans to keep the system properly cooled.	

## 5.2 Temperature Sensors

There are a variety of temperature sensors that can be implemented on Intel® Server Systems. They are split into three types: regular temperature sensors, thermal margin sensors, and discrete temperature sensors. Each of them has its own types of events that can be logged.

### 5.2.1 Regular Temperature Sensors

Regular temperature sensors are sensors that report an actual temperature. These are linear, threshold-based sensors. In most Intel® Server Systems, there are at least two sensors defined: front panel temperature and baseboard temperature. Both these sensors typically have upper and lower thresholds set – upper to warn in case of an over-temperature situation, lower to warn against sensor failure (temperature sensors typically read out 0 if they stop working).

Table 33: Temperature Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	01h = Temperature
12	Sensor Number	See Table 35
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Trigger Offset as described in Table 34
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

Table 34: Temperature Sensors Event Triggers – Description

Event Trigger		Assertion Severity	Deassert Severity	Description
Hex	Description			
00h	Lower non-critical going low	Degraded	OK	The temperature has dropped below its lower non-critical threshold.
02h	Lower critical going low	non-fatal	Degraded	The temperature has dropped below its lower critical threshold.
07h	Upper non-critical going high	Degraded	OK	The temperature has gone over its upper non-critical threshold.
09h	Upper critical going high	non-fatal	Degraded	The temperature has gone over its upper critical threshold.

Table 35: Temperature Sensors – Next Steps

Sensor Name	Sensor Number	Next Steps
Baseboard Temp	20h	<ol style="list-style-type: none"> <li>1. Check for clear and unobstructed airflow into and out of the chassis.</li> <li>2. Ensure the SDR is programmed and correct chassis has been selected.</li> <li>3. Ensure there are no fan failures.</li> <li>4. Ensure the air used to cool the system is within the thermal specifications for the system (typically below 35°C).</li> </ol>
Front Panel Temp	21h	<p>If the front panel temperature reads zero, check:</p> <ol style="list-style-type: none"> <li>1. It is connected properly.</li> <li>2. The FRUSDR has been programmed correctly for your chassis.</li> </ol> <p>If the front panel temperature is too high:</p> <ul style="list-style-type: none"> <li>▪ Check the cooling of your server room.</li> </ul>

## 5.2.2 Thermal Margin Sensors

Margin sensors are also linear sensors but typically report a negative value. This is not an actual temperature, but in fact an offset to a critical temperature. Example sensors are Processor Thermal Margin, Memory Thermal Margin, and IOH Thermal margin. Values reported should be seen as number of degrees below a critical temperature for the particular component.

Table 36: Thermal Margin Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	01h = Temperature
12	Sensor Number	See Table 38
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Triggers as described in Table 37
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

Table 37: Thermal Margin Sensors Event Triggers – Description

Event Trigger		Assertion Severity	Deassert Severity	Description
Hex	Description			
07h	Upper non-critical going high	Degraded	OK	The thermal margin has gone over its upper non-critical threshold.
09h	Upper critical going high	non-fatal	Degraded	The thermal margin has gone over its upper critical threshold.

Table 38: Thermal Margin Sensors – Next Steps

Sensor Number	Sensor Name	Next Steps
22h	IOH Therm Margin	<ol style="list-style-type: none"> <li>1. Check for clear and unobstructed airflow into and out of the chassis.</li> <li>2. Ensure the SDR is programmed and correct chassis has been selected.</li> <li>3. Ensure there are no fan failures.</li> <li>4. Ensure the air used to cool the system is within the thermal specifications for the system (typically below 35°C).</li> </ol>
23h	Mem P1 Therm Margin	
24h	Mem P2 Therm Margin	
62h	P1 Therm Margin	Not a logged SEL event. Sensor is used for thermal management of the processor.
63h	P2 Therm Margin	

### 5.2.3 Processor Thermal Control % Sensors

Processor Thermal Control % sensors report the percentage of the time that the processor is throttling its performance due to thermal issues. If this is not addressed the processor could overheat and shut down the system to protect itself from damage.

Table 39: Processor Thermal Control % Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	01h = Temperature
12	Sensor Number	See Table 41
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Triggers as described in Table 40
15	Event Data 2	Reading that triggered event.

Byte	Field	Description
16	Event Data 3	Threshold value that triggered event.

Table 40: Processor Thermal Control % Sensors Event Triggers – Description

Event Trigger		Assertion Severity	Deassert Severity	Description
Hex	Description			
07h	Upper non-critical going high	Degraded	OK	The thermal margin has gone over its upper non-critical threshold.
09h	Upper critical going high	non-fatal	Degraded	The thermal margin has gone over its upper critical threshold.

Table 41: Processor Thermal Control % Sensors – Next Steps

Sensor Number	Sensor Name	Next Steps
64h	P1 Therm Ctl %	These events normally only happen due to failures of the thermal solution: <ol style="list-style-type: none"> <li>1. Verify the heatsink is properly attached and has thermal grease.</li> <li>2. If the system has a heatsink fan, ensure the fan is spinning.</li> <li>3. Check all system fans are operating properly.</li> <li>4. Check that the air used to cool the system is within limits (typically 35°C).</li> </ol>
65h	P2 Therm Ctl %	

## 5.2.4 Discrete Thermal Sensors

Discrete thermal sensors do not report a temperature at all – instead they report an overheating event of some kind. Examples as VRD Hot (voltage regulator is overheating) or processor Thermal Trip (the processor got so hot that its over-temperature protection was triggered and the system was shut down to prevent damage).

Table 42: Discrete Thermal Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	01h = Temperature
12	Sensor Number	See Table 43
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = See Table 43
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 43
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 43: Discrete Thermal Sensors – Next Steps

Sensor Number	Sensor Name	Event Type	Event Trigger Offset		Description	Next Steps
			Hex	Description		
66h	P1 VRD Hot	05h	01h	Limit Exceeded	Processor1 voltage regulator overheated	<ol style="list-style-type: none"> <li>1. Check for clear and unobstructed airflow into and out of the chassis.</li> <li>2. Ensure the SDR is programmed and correct chassis has been selected.</li> <li>3. Ensure there are no fan failures.</li> <li>4. Ensure the air used to cool the system is within the thermal specifications for the system (typically below 35°C).</li> </ol>
67h	P2 VRD Hot				Processor2 voltage regulator overheated	
6ah	IOH Thermal Trip	03h	01h	State Asserted	I/O Hub (IOH) overheated	



## 6. Processor Subsystem

Intel® servers report several processor-centric sensors in the SEL.

### 6.1 Processor Status Sensor

The status sensor reports processor presence or a thermal trip condition. Each processor has a status sensor.

Table 44: Process Status Sensors Typical Characteristics

Byte	Field	Description
11	Sensor Type	07h = Processor
12	Sensor Number	See Table 45
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 45
15	Event Data 2	Not used.
16	Event Data 3	Not used.

Table 45: Processor Status Sensors – Next Steps

Sensor Number	Sensor Name	Event Trigger Offset		Description	Next Steps
		Hex	Description		
60h	P1 Status	01h	Thermal trip	The processor exceeded the maximum temperature.	This event normally only happens due to failures of the thermal solution: <ol style="list-style-type: none"> <li>1. Verify the heatsink is properly attached and has thermal grease.</li> <li>2. If the system has a heatsink fan, ensure the fan is spinning.</li> <li>3. Check all system fans are operating properly.</li> <li>4. Check that the air used to cool the system is within limits (typically 35°C).</li> </ol>
		07h	State Asserted	Indicates the processor is present.	
61h	P2 Status	01h	Thermal trip	The processor exceeded the maximum temperature.	
		07h	State Asserted	Indicates the processor is present.	

## 6.2 Catastrophic Error Sensor

When the Catastrophic Error signal (CATERR#) stays asserted, it is a sign that something serious has gone wrong in the hardware. The BMC monitors this signal and reports when it stays asserted.

Table 46: Catastrophic Error Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	07h = Processor
12	Sensor Number	68h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 01h (State Asserted)
15	Event Data 2	Not used.

Byte	Field	Description
16	Event Data 3	Not used.

### 6.2.1 Catastrophic Error Sensor – Next Steps

This error is typically caused by other platform components.

1. Check for other errors near the time of the CATERR event.
2. Verify all peripherals are plugged in and operating correctly, particularly Hard Drives, Optical Drives, and I/O.
3. Update system firmware and drivers.

## 6.3 CPU Missing Sensor

The CPU Missing sensor is a discrete sensor reporting the processor is not installed. The most common instance of this event is due to a processor populated in the incorrect socket.

Table 47: CPU Missing Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	07h = Processor
12	Sensor Number	69h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 01h (State Asserted)
15	Event Data 2	Not used.
16	Event Data 3	Not used.

### 6.3.1 CPU Missing Sensor – Next Steps

Verify the processor is installed in the correct slot.

## 6.4 QuickPath Interconnect Error Sensors

The Intel® QuickPath Interconnect (QPI) bus on Intel® S5500/S3420 series server boards is the interconnection between processors and to the chipset. The QPI Error sensors are all reported by the BIOS SMI Handler to the BMC so the Generator ID will be 33h.

### 6.4.1 QPI Correctable Error Sensor

The system detected an error and corrected it. This is an informational event.

Table 48: QPI Correctable Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	06h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 72h (OEM Discrete)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = Reserved
15	Event Data 2	0-3 = CPU1-4
16	Event Data 3	Not used

### 6.4.1.1 QPI Correctable Error Sensor – Next Steps

This is an Informational event only. Correctable errors are acceptable and normal at a low rate of occurrence. If the error continues:

1. Check the processor is installed correctly.
2. Inspect the socket for bent pins.
3. Cross test the processor if possible.

### 6.4.2 QPI Non-Fatal Error Sensor

The system detected a QPI non-fatal error that is recoverable. This is an informational event.

Table 49: QPI Non-Fatal Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	07h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 73h (OEM Discrete)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = Reserved
15	Event Data 2	0-3 = CPU1-4
16	Event Data 3	Not used

### 6.4.2.1 QPI Non-Fatal Error Sensor – Next Steps

This is an Informational event only. Non-Fatal errors are acceptable and normal at a low rate of occurrence. If the error continues:

1. Check the processor is installed correctly.
2. Inspect the socket for bent pins.
3. Cross test the processor if possible.

### 6.4.3 QPI Fatal and Fatal #2

The system detected a QPI fatal or non-recoverable error. This is a fatal error.

Table 50: QPI Fatal Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	17h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 74h (OEM Discrete)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = Reserved
15	Event Data 2	0-3 = CPU1-4
16	Event Data 3	Not used

The QPI Fatal #2 Error is a continuation of QPI Fatal Error.

Table 51: QPI Fatal #2 Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	18h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 74h (OEM Discrete)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = Reserved
15	Event Data 2	0-3 = CPU1-4
16	Event Data 3	Not used

#### 6.4.3.1 QPI Fatal and Fatal #2 – Next Steps

This is an Informational event only. Correctable errors are acceptable and normal at a low rate of occurrence. If the error continues:

1. Check the processor is installed correctly.
2. Inspect the socket for bent pins.
3. Cross test the processor if possible.

## 7. Memory Subsystem

Intel® servers report memory errors, status, and configuration in the SEL.

### 7.1 Memory RAS Mirroring and Sparing

“Memory RAS Configuration Status” refers to the BIOS sending the current RAS mode and RAS operational state to the BMC to log into the SEL as a SEL record. This allows a remote software/application to query and retrieve the system memory state.

The memory configuration state sensors are “virtual” sensors. In other words, these sensors are owned and controlled completely by the BIOS, independently of the BMC.

The RAS configuration and state definitions are aligned with the definitions within the *Intelligent Platform Management Interface Specification*, Version 2.0. Accordingly, these sensors are read as “Status” and “Redundancy” sensors (Event/Reading Type 0x09 and 0x0B respectively).

- Sensor Number 12h (Event Type 0x09) – Mirroring Configuration Status
- Sensor Number 01h (Event Type 0x0B) – Mirroring Redundancy State
- Sensor Number 13h (Event Type 0x09) – Sparing Configuration Status
- Sensor Number 11h (Event Type 0x0B) – Sparing Redundancy State

#### 7.1.1 Mirroring Configuration Status

This sensor provides the Mirroring mode RAS configuration status.

Table 52: Mirroring Configuration Status Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0001h = BIOS POST
11	Sensor Type	0ch = Memory
12	Sensor Number	12h



Byte	Field	Description
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 09h (digital Discrete)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 53
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 53: Mirroring Configuration Status Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
01h	The system has been configured into Mirrored Channel RAS Mode.	User enabled mirrored channel mode in setup.	Informational event only.
00h	The system has been configured out of Mirrored Channel RAS Mode.	Mirrored channel mode is disabled (either in setup or due to unavailability of memory at post, in which case post error 8500 is also logged).	<ol style="list-style-type: none"> <li>1. If this event is accompanied by a post error 8500, there was a problem applying the mirroring configuration to the memory. Check for other errors related to the memory and troubleshoot accordingly.</li> <li>2. If there is no post error then mirror mode was simply disabled in BIOS setup and this should be considered informational only.</li> </ol>

### 7.1.2 Mirrored Redundancy State Sensor

This sensor provides the RAS Redundancy state for the Memory Mirrored Channel Mode.

Table 54: Mirrored Redundancy State Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0001h = BIOS POST
11	Sensor Type	0ch = Memory
12	Sensor Number	01h
13	Event Direction and Event Type	<p>[7] Event direction  0b = Assertion Event  1b = Deassertion Event</p> <p>[6:0] Event Type = 0Bh (Generic Discrete)</p>
14	Event Data 1	<p>[7:6] – 10b = OEM code in Event Data 2  [5:4] – 10b = OEM code in Event Data 3  [3:0] – Event Trigger Offset as described in Table 55</p>
15	Event Data 2	<p>[7:4] – If Domain Instance Type (ED3) is set to Local, this field specifies the mirroring domain local sub-instances – which channels are included in this sub-instance:  0000b – Reserved  0001b – {Ch A, Ch B}  0010b – {Ch A, Ch C}  0011b – {Ch B, Ch C}  0100b-1110b – Reserved</p> <p>If Domain Instance Type (ED3) is set to Global, this field specifies the 0-based Socket ID of the first participant processor in this mirroring domain global instance.  A value of 1111b indicates that this field is unused and does not contain valid data.</p> <p>[3:0] – If Domain Instance Type (ED3) is set to Local, this field specifies the sparing domain local sub-instances – which channels are included in this sub-instance:  0000b – Reserved  0001b – {Ch A, Ch B, Ch C} (only configuration possible on Intel® S5500/S5520 Server Boards)  0010b-1110b – Reserved</p> <p>If Domain Instance Type (ED3) is set to Global, this field specifies the 0-based Socket ID of the first participant processor in this sparing domain global instance.  A value of 1111b indicates that this field is unused and does not contain valid data.</p>

Byte	Field	Description
16	Event Data 3	<p>[7] – Domain Instance Type</p> <p>0b: Local memory sparing domain instance. This SEL pertains to a local memory mirroring domain that is restricted to memory mirroring pairs within a processor socket only.</p> <p>1b: Global memory sparing domain instance. This SEL pertains to a global memory mirroring domain that pertains to memory mirroring between processor sockets.</p> <p>[6:4] – Reserved</p> <p>[3:0] – 0-based Instance ID of this sparing domain</p>

Table 55: Mirrored Redundancy State Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
01h	Memory is configured in Mirrored Channel Mode, and the memory is operating in the fully redundant state.	System boots with mirrored channel mode active, one entry per processor.	Informational event.
00h	Memory is configured in Mirrored Channel Mode, and the memory has lost redundancy and is operating in the degraded state.	One of the channels in the mirror pair is taken offline – loss of mirror – one entry only for affected processor.	This event should be accompanied by memory errors indicating the source of the issue. Troubleshoot accordingly (probably replace affected DIMM).

### 7.1.3 Sparing Configuration Status

This sensor provides the Spare Channel mode RAS Configuration status.

Table 56: Sparing Configuration Status Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0001h = BIOS POST
11	Sensor Type	0ch = Memory

Byte	Field	Description
12	Sensor Number	13h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 09h (digital Discrete)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 57
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 57: Sparing Configuration Status Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
01h	The system has configured into Spare Channel RAS mode.	Sparing mode is enabled in setup.	Informational event only.
00h	The system has configured out of Spare Channel RAS mode	Sparing mode is disabled, either from setup or due to error in which case post error 8500 also occurs.	<ol style="list-style-type: none"> <li>1. If this event is accompanied by a post error 8500, there was a problem applying the sparing configuration to the memory. Check for other errors related to the memory and troubleshoot accordingly.</li> <li>2. If there is no post error then sparing mode was simply disabled in BIOS setup and this should be considered informational only.</li> </ol>

### 7.1.4 Sparing Redundancy State Sensor

This sensor provides the RAS Redundancy state for the Spare Channel Mode.

Table 58: Sparing Redundancy State Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0001h = BIOS POST
11	Sensor Type	0ch = Memory
12	Sensor Number	11h
13	Event Direction and Event Type	<p>[7] Event direction  0b = Assertion Event  1b = Deassertion Event</p> <p>[6:0] Event Type = 0Bh (Generic Discrete)</p>
14	Event Data 1	<p>[7:6] – 10b = OEM code in Event Data 2  [5:4] – 10b = OEM code in Event Data 3  [3:0] – Event Trigger Offset as described in Table 59</p>
15	Event Data 2	<p>[7:4] – If Domain Instance Type (ED3) is set to Local, this field specifies the 0-based Socket ID of the processor that contains the sparing domain local sub-instances.  A value of 1110b indicates that the sparing configuration specified in Bits [3:0] applies globally to all sockets in the system.  If Domain Instance Type (ED3) is set to Global, this field specifies the 0-based Socket ID of the second participant processor in this sparing domain global instance.  A value of 1111b indicates that this field is unused and does not contain valid data.</p> <p>[3:0] – If Domain Instance Type (ED3) is set to Local, this field specifies the sparing domain local sub-instances – which channels are included in this sub-instance:  0000b – Reserved  0001b – {Ch A, Ch B, Ch C} (only configuration possible on Intel® S5500/S5520 Server Boards)  0010b-1110b – Reserved</p> <p>If Domain Instance Type (ED3) is set to Global, this field specifies the 0-based Socket ID of the first participant processor in this sparing domain global instance.  A value of 1111b indicates that this field is unused and does not contain valid data.</p>

Byte	Field	Description
16	Event Data 3	<p>[7] – Domain Instance Type</p> <p>0b: Local memory sparing domain instance. This SEL pertains to a local memory sparing domain that is restricted to memory sparing pairs within a processor socket only.</p> <p>1b: Global memory sparing domain instance. This SEL pertains to a global memory sparing domain that pertains to memory sparing between processor sockets.</p> <p>[6:4] – Reserved</p> <p>[3:0] – 0-based Instance ID of this sparing domain</p>

Table 59: Sparing Redundancy State Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
01h	Memory is configured in Spare Channel Mode, and the memory is operating in the fully redundant state, with the spare channel inactive and available.	System boots with spare channel mode active, one entry per processor.	Informational event.
00h	Memory is configured in Spare Channel Mode, and the memory has lost redundancy and is operating in the degraded state, with the spare channel active and used to replace a failed channel.	Spare channel replaces failing channel, one SEL entry for processor with failing memory to signify loss of redundancy.	This event should be accompanied by memory errors indicating the source of the issue. Troubleshoot accordingly (probably replace affected DIMM).

## 7.2 ECC and Address Parity

1. Memory data errors are logged as correctable or uncorrectable.
2. Uncorrectable errors are fatal.
3. Memory addresses are protected with parity bits and a parity error is logged. This is a fatal error.

### 7.2.1 Memory Correctable and Uncorrectable ECC Error

ECC errors are divided into Uncorrectable ECC Errors and Correctable ECC Errors. A “Correctable ECC Error” actually represents a threshold overflow. More Correctable Errors are detected at the memory controller level for a given DIMM within a given timeframe. In both cases, the error can be narrowed down to particular DIMM(s). The BIOS SMI error handler uses this information to log the data to the BMC SEL and identify the failing DIMM module.

Table 60: Correctable and Uncorrectable ECC Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	0ch = Memory
12	Sensor Number	02h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Event Trigger Offset as described in Table 61
15	Event Data 2	[7:2] – Reserved. Set to 0. [1:0] – The logical rank associated with the failed DDR3 DIMM

Byte	Field	Description
16	Event Data 3	<p>[7:5] – Indicates the Processor Socket to which the DDR3 DIMM having the ECC error is attached:</p> <p>000b = Processor Socket 1</p> <p>001b = Processor Socket 2</p> <p>All other values are reserved.</p> <p>[4:3] – Indicates the processor Memory Channel to which the failing DDR3 DIMM is attached:</p> <p>00b = Channel A or D (For Processor Socket 1, Processor Socket 2)</p> <p>01b = Channel B or E</p> <p>10b = Channel C or F</p> <p>11b is reserved.</p> <p>[2:0] – Indicates the DIMM Socket on the channel to which the failing DDR3 DIMM is attached:</p> <p>000b = DIMM Socket 1</p> <p>001b = DIMM Socket 2</p> <p>All other values are reserved.</p>

Table 61: Correctable and Uncorrectable ECC Error Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
01h	Uncorrectable ECC Error	<p>An uncorrectable (multi-bit) ECC error has occurred. This is a fatal issue that will typically lead to an OS crash (unless memory has been configured in a RAS mode). The system will generate a CATERR# (catastrophic error) and an MCE (Machine Check Exception Error).</p> <p>While the error may be due to a failing DRAM chip on the DIMM, it could also be caused by incorrect seating or improper contact between the socket and DIMM, or by bent pins in the processor socket.</p>	<ol style="list-style-type: none"> <li>1. If needed, decode DIMM location from hex version of SEL.</li> <li>2. Verify the DIMM is seated properly.</li> <li>3. Examine gold fingers on edge of the DIMM to verify contacts are clean.</li> <li>4. Inspect the processor socket this DIMM is connected to for bent pins, and if found, replace the board.</li> <li>5. Consider replacing the DIMM as a preventative measure. For multiple occurrences, replace the DIMM.</li> </ol>



Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Correctable ECC Error threshold reached	There have been too many (10 or more) correctable ECC errors for this particular DIMM since last boot. This event in itself does not pose any direct problems as the ECC errors are still being corrected. Depending on the RAS configuration of the memory, the IMC may take the affected DIMM offline.	<p>Even though this event doesn't immediately lead to problems, it can indicate one of the DIMM modules is slowly failing. If this error occurs more than once:</p> <ol style="list-style-type: none"> <li>1. If needed, decode DIMM location from hex version of SEL.</li> <li>2. Verify the DIMM is seated properly.</li> <li>3. Examine gold fingers on edge of the DIMM to verify contacts are clean.</li> <li>4. Inspect the processor socket this DIMM is connected to for bent pins, and if found, replace the board.</li> <li>5. Consider replacing the DIMM as a preventative measure. For multiple occurrences, replace the DIMM.</li> </ol>

## 7.2.2 Memory Address Parity Error

Address Parity errors are errors detected in the memory addressing hardware. Because these affect the addressing of memory contents, they can potentially lead to the same sort of failures as ECC errors. They are logged as a distinct type of error because they affect memory addressing rather than memory contents, but otherwise they are treated exactly the same as Uncorrectable ECC Errors. Address Parity errors are logged to the BMC SEL, with Event Data to identify the failing address by channel and DIMM to the extent that it is possible to do so.

Table 62: Address Parity Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	0ch = Memory
12	Sensor Number	14h
13	Event Direction and Event Type	<p>[7] Event direction  0b = Assertion Event  1b = Deassertion Event</p> <p>[6:0] Event Type = 6Fh (Sensor Specific)</p>

Byte	Field	Description
14	Event Data 1	<p>[7:6] – 10b = OEM code in Event Data 2</p> <p>[5:4] – 10b = OEM code in Event Data 3</p> <p>[3:0] – Event Trigger Offset = 02h</p>
15	Event Data 2	<p>[7:5] – Reserved. Set to 0.</p> <p>[4] – Channel Information Validity Check:  0b = Channel Number in Event Data 3 Bits[4:3] is not valid  1b = Channel Number in Event Data 3 Bits[4:3] is valid</p> <p>[3] – DIMM Information Validity Check:  0b = DIMM Slot ID in Event Data 3 Bits[2:0] is not valid  1b = DIMM Slot ID in Event Data 3 Bits[2:0] is valid</p> <p>[2:0] – Error Type:  000b = Parity Error Type not known  001b = Data Parity Error (not used)  010b = Address Parity Error  All other values are reserved.</p>
16	Event Data 3	<p>[7:5] – Indicates the Processor Socket to which the DDR3 DIMM having the ECC error is attached:  000b = Processor Socket 1  001b = Processor Socket 2  All other values are reserved.</p> <p>[4:3] – Channel Number (if valid) on which the Parity Error occurred. This value will be indeterminate and should be ignored if ED2 Bit [4] is 0b.  00b = Channel A or D (For Processor Socket 1, Processor Socket 2)  01b = Channel B or E  10b = Channel C or F  11b = Reserved</p> <p>[2:0] – DIMM Slot ID (if valid) of the specific DIMM that was involved in the transaction that led to the parity error. This value will be indeterminate and should be ignored if ED2 Bit [3] is 0b.  000b = DIMM Socket 1  001b = DIMM Socket 2  All other values are reserved.</p>

### 7.2.2.1 Memory Address Parity Error Sensor Next Steps

These are bit errors that are detected in the memory addressing hardware. An Address Parity Error implies that the memory address transmitted to the DIMM addressing circuitry has been compromised, and data read or written is compromised in turn. An Address Parity Error is logged as such in SEL but in all other ways is treated the same as an Uncorrectable ECC Error.

While the error may be due to a failing DRAM chip on the DIMM, it could also be caused by incorrect seating or improper contact between the socket and DIMM, or by bent pins in the processor socket.

1. If needed, decode DIMM location from hex version of SEL.
2. Verify the DIMM is seated properly.
3. Examine gold fingers on edge of the DIMM to verify contacts are clean.
4. Inspect the processor socket this DIMM is connected to for bent pins, and if found, replace the board.
5. Consider replacing the DIMM as a preventative measure. For multiple occurrences, replace the DIMM.

## 8. PCI Express\* and Legacy PCI Subsystem

The *PCI Express\* (PCIe) Specification* defines standard error types under the Advanced Error Reporting (AER) capabilities. The BIOS logs AER events into the SEL.

The *Legacy PCI Specification* error types are PERR and SERR. These errors are supported and logged into the SEL.

### 8.1 PCI Express\* Errors

PCIe error events are either correctable (informational event) or fatal. In both cases information is logged to help identify the source of the PCIe error and the bus, device, and function is included in the extended data fields. The PCIe devices are mapped in the operating system by bus, device, and function. Each device is uniquely identified by the bus, device, and function. PCIe device information can be found in the operating system.

#### 8.1.1 PCI Express\* Correctable Errors

When a PCI Express\* correctable error is reported to the BIOS SMI handler, it will record the error using the following format.

Table 63: PCI Express\* Correctable Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	05h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 71h (OEM Specific)

Byte	Field	Description
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Event Trigger Offset as described in Table 64
15	Event Data 2	PCI Bus number
16	Event Data 3	[7:3] – PCI Device number [2:0] – PCI Function number

Table 64: PCI Express\* Correctable Error Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Receiver error	Correctable error occurred	Informational event only. Correctable errors are acceptable and normal at a low rate of occurrence. If the error continues: <ol style="list-style-type: none"> <li>1. Decode bus, device, and function to identify the card.</li> <li>2. If this is an add-in card:               <ol style="list-style-type: none"> <li>a. Verify the card is inserted properly.</li> <li>b. Install the card in another slot and check whether the error follows the card or stays with the slot.</li> <li>c. Update all firmware and drivers, including non-Intel components.</li> </ol> </li> <li>3. If this is an onboard device:               <ol style="list-style-type: none"> <li>a. Update all BIOS, firmware, and drivers.</li> <li>b. Replace the board.</li> </ol> </li> </ol>
01h	Bad DLLP error	Correctable bad DLLP occurred	
02h	Bad TLLP error	Correctable bad TLP occurred	
03h	REPLAY_NUM Rollover Error	Correctable Replay event occurred	
04h	REPLAY Timer Timeout Error	Correctable Replay timeout event occurred	
05h	Advisory non-fatal Error (received ERR_COR message)	Correctable advisory event occurred, typically provided as notice to software driver	
06h	Link bandwidth changed	Link bandwidth changed	

## 8.1.2 PCI Express\* Fatal Errors

When a PCI Express\* fatal error is reported to the BIOS SMI handler, it will record the error using the following format.

Table 65: PCI Express\* Fatal Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	04h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 70h (OEM Specific)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Event Trigger Offset as described in Table 66
15	Event Data 2	PCI Bus number
16	Event Data 3	[7:3] – PCI Device number [2:0] – PCI Function number

Table 66: PCI Express\* Fatal Error Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Data Link Layer Protocol Error	Indicates a CRC error detected during a DLLP transaction. This means the transaction was corrupted.	<ol style="list-style-type: none"> <li>Decode bus, device, and function to identify the card.</li> <li>If this is an add-in card: <ol style="list-style-type: none"> <li>Verify the card is inserted properly.</li> <li>Install the card in another slot and check whether the error follows the card or stays with the slot.</li> <li>Update all firmware and drivers, including non-Intel components.</li> </ol> </li> </ol>
01h	Surprise Link Down	The link was lost and is no longer functional. Requires a reboot to bring the link back.	
02h	Unexpected Completion	Indicates the device received a completion notification for a transaction it does not recognize. This is a fatal error.	
03h	Received Unsupported request condition on inbound address decode with the exception of SAD	Typically indicates a failure due to an incorrect address sent to the target. This unknown address is a fatal error.	

Event Trigger Offset		Description	Next Steps
Hex	Description		
04h	Poisoned TLP Error	Typically indicates a parity error in a TLP transaction. This means the data received is not correct.	
05h	Flow Control Protocol Error	Indicates an error during initialization with the device not providing enough flow control credits. This means the bus configuration is incorrect and it cannot continue.	
06h	Completion Timeout Error	Indicates a transaction did not complete in the specified amount of time.	
07h	Completer Abort Error	Indicates a transaction had unexpected content or format.	
08h	Receiver Buffer Overflow Error	Indicates a synchronization problem between PCI Express* devices. Extremely rare.	
09h	ACS Violation Error	Access Control Services, a transaction routing feature, failed.	
0Ah	Malformed TLP Error	Indicates a transaction was sent with data exceeding the maximum allowed number of bytes. This is not allowed and is a fatal error, usually a firmware or driver problem.	
0Bh	Received ERR_FATAL message from downstream Error	Indicates a fatal error occurred and is being reported.	
0Ch	Unexpected Completion Error	Indicates the device received a completion notification for a transaction it does not recognize.	
0Dh	Received ERR_NONFATAL Message Error	Indicates a non-fatal error is redefined as fatal, and is being reported.	

3. If this is an onboard device:
- Update all BIOS, firmware, and drivers.
  - Replace the board.

### 8.1.3 Legacy PCI Errors

Legacy PCI errors include PERR and SERR; both are fatal errors.

Table 67: Legacy PCI Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0033h = BIOS SMI Handler
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	03h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Event Trigger Offset as described in Table 68
15	Event Data 2	PCI Bus number
16	Event Data 3	[7:3] – PCI Device number [2:0] – PCI Function number

Table 68: Legacy PCI Error Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
04h	PERR#	Parity Error, PERR, asserted. This is a fatal error.	<ol style="list-style-type: none"> <li>1. Decode bus, device, and function to identify the card.</li> <li>2. If this is an add-in card: <ol style="list-style-type: none"> <li>a. Verify the card is inserted properly.</li> <li>b. Install the card in another slot and check whether the error follows the card or stays with the slot.</li> <li>c. Update all firmware and drivers, including non-Intel components.</li> </ol> </li> <li>3. If this is an onboard device: <ol style="list-style-type: none"> <li>a. Update all BIOS, firmware, and drivers.</li> <li>b. Replace the board.</li> </ol> </li> </ol>
05h	SERR#	System Error, SERR, asserted. This is a fatal error.	



## 9. System BIOS Events

---

There are a number of events that are owned by the system BIOS. These events can occur during Power On Self Test (POST) or when coming out of a sleep state. Not all of these events signify errors. Some events are described in other chapters in this document (for example, memory events).

### 9.1 System Events

These events can occur during POST or when coming out of a sleep state. These are informational events only.

1. When logging events during POST BIOS uses generator ID 0001h.
2. When coming out of a sleep state BIOS uses generator ID 0033h.

#### 9.1.1 System Boot

The BIOS logs a system boot event every time the system boots. The event gets logged early during POST when BIOS-BMC communication is first established. This event is not an error.

#### 9.1.2 Timestamp Clock Synchronization

These events are used when the time between the BIOS and the BMC is synchronized. Two events are logged. The BIOS does the first one to send the time synch message to the BMC for synchronization, and the timestamp that the message gets is unknown, that is, the timestamp in the log could be anything because it gets the "before" timestamp.

So the BIOS sends a second time synch message to get a "baseline" correct timestamp in the log. That is the "starting time".

For example, say that the time the BMC has is March 1, 2011 21:00. The BIOS time synch updates that to the same date, 21:20 (the BMC was running behind). Without that second time synch message, you don't know that the log time jumped ahead, and when you get the next log message it looks like there was a 20-min delay during the boot for some unknown reasons.

Without that second time synch message, the time span to the next logged message is indeterminate. With the second time synch as a baseline, the following log timestamps are always determinate.

Table 69: System Event Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	<ul style="list-style-type: none"> <li>0001h = BIOS POST</li> <li>0033h = BIOS SMI Handler</li> </ul>
11	Sensor Type	12h = System Event
12	Sensor Number	83h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset 01h = System Boot 05h = Timestamp Clock Synchronization
15	Event Data 2	For Event Trigger Offset 05h only (Timestamp Clock Synchronization) 00h = 1st in pair 80h = 2nd in pair
16	Event Data 3	Not used

## 9.2 System Firmware Progress (Formerly Post Error)

The BIOS logs any POST errors to the SEL. The 2-byte POST code gets logged in the ED2 and ED3 bytes in the SEL entry. This event will be logged every time a POST error is displayed. Even though this event indicates an error, it may not be a fatal error. If this is a serious error, there will typically also be a corresponding SEL entry logged for whatever was the cause of the error – this event may contain more information about what happened than the POST error event.

Table 70: POST Error Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0001h = BIOS POST
11	Sensor Type	0Fh = System Firmware Progress (formerly POST Error)
12	Sensor Number	06h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Event Trigger Offset = 0
15	Event Data 2	Low Byte of POST Error Code
16	Event Data 3	High Byte of POST Error Code

### 9.2.1 System Firmware Progress (Formerly Post Error) - Next Steps

See the following table for POST error Codes.

Table 71: POST Error Codes

Error Code	Error Message	Response
0012	CMOS date/time not set	Major
0048	Password check failed	Major
0108	Keyboard component encountered a locked error.	Minor
0109	Keyboard component encountered a stuck key error.	Minor

Error Code	Error Message	Response
0113	Fixed Media The SAS RAID firmware cannot run properly. The user should attempt to reflash the firmware.	Major
0140	PCI component encountered a PERR error.	Major
0141	PCI resource conflict	Major
0146	PCI out of resources error	Major
0192	Processor 0x cache size mismatch detected.	Fatal
0193	Processor 0x stepping mismatch.	Minor
0194	Processor 0x family mismatch detected.	Fatal
0195	Processor 0x Intel® QPI speed mismatch.	Fatal
0196	Processor 0x model mismatch.	Fatal
0197	Processor 0x speeds mismatched.	Fatal
0198	Processor 0x family is not supported.	Fatal
019F	Processor and chipset stepping configuration is unsupported.	Major
5220	CMOS/NVRAM Configuration Cleared	Major
5221	Passwords cleared by jumper	Major
5224	Password clear Jumper is Set.	Major
8160	Processor 01 unable to apply microcode update	Major
8161	Processor 02 unable to apply microcode update	Major
8180	Processor 0x microcode update not found.	Minor
8190	Watchdog timer failed on last boot	Major
8198	OS boot watchdog timer failure.	Major
8300	Baseboard management controller failed self-test	Major
84F2	Baseboard management controller failed to respond	Major
84F3	Baseboard management controller in update mode	Major
84F4	Sensor data record empty	Major
84FF	System event log full	Minor

Error Code	Error Message	Response
8500	Memory component could not be configured in the selected RAS mode.	Major
8501	DIMM Population Error.	Major
8502	CLTT Configuration Failure Error.	Major
8520	DIMM_A1 failed Self-Test (BIST).	Major
8521	DIMM_A2 failed Self-Test (BIST).	Major
8522	DIMM_B1 failed Self-Test (BIST).	Major
8523	DIMM_B2 failed Self-Test (BIST).	Major
8524	DIMM_C1 failed Self-Test (BIST).	Major
8525	DIMM_C2 failed Self-Test (BIST).	Major
8526	DIMM_D1 failed Self-Test (BIST).	Major
8527	DIMM_D2 failed Self-Test (BIST).	Major
8528	DIMM_E1 failed Self-Test (BIST).	Major
8529	DIMM_E2 failed Self-Test (BIST).	Major
852A	DIMM_F1 failed Self-Test (BIST).	Major
852B	DIMM_F2 failed Self-Test (BIST).	Major
8540	DIMM_A1 Disabled.	Major
8541	DIMM_A2 Disabled.	Major
8542	DIMM_B1 Disabled.	Major
8543	DIMM_B2 Disabled.	Major
8544	DIMM_C1 Disabled.	Major
8545	DIMM_C2 Disabled.	Major
8546	DIMM_D1 Disabled.	Major
8547	DIMM_D2 Disabled.	Major
8548	DIMM_E1 Disabled.	Major
8549	DIMM_E2 Disabled.	Major

Error Code	Error Message	Response
854A	DIMM_F1 Disabled.	Major
854B	DIMM_F2 Disabled.	Major
8560	DIMM_A1 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8561	DIMM_A2 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8562	DIMM_B1 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8563	DIMM_B2 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8564	DIMM_C1 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8565	DIMM_C2 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8566	DIMM_D1 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8567	DIMM_D2 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8568	DIMM_E1 Component encountered a Serial Presence Detection (SPD) fail error.	Major
8569	DIMM_E2 Component encountered a Serial Presence Detection (SPD) fail error.	Major
856A	DIMM_F1 Component encountered a Serial Presence Detection (SPD) fail error.	Major
856B	DIMM_F2 Component encountered a Serial Presence Detection (SPD) fail error.	Major
85A0	DIMM_A1 Uncorrectable ECC error encountered.	Major
85A1	DIMM_A2 Uncorrectable ECC error encountered.	Major
85A2	DIMM_B1 Uncorrectable ECC error encountered.	Major
85A3	DIMM_B2 Uncorrectable ECC error encountered.	Major
85A4	DIMM_C1 Uncorrectable ECC error encountered.	Major
85A5	DIMM_C2 Uncorrectable ECC error encountered.	Major
85A6	DIMM_D1 Uncorrectable ECC error encountered.	Major
85A7	DIMM_D2 Uncorrectable ECC error encountered.	Major
85A8	DIMM_E1 Uncorrectable ECC error encountered.	Major
85A9	DIMM_E2 Uncorrectable ECC error encountered.	Major
85AA	DIMM_F1 Uncorrectable ECC error encountered.	Major

Error Code	Error Message	Response
85AB	DIMM_F2 Uncorrectable ECC error encountered.	Major
8604	Chipset Reclaim of non-critical variables complete.	Minor
9000	Unspecified processor component has encountered a non-specific error.	Major
9223	Keyboard component was not detected.	Minor
9226	Keyboard component encountered a controller error.	Minor
9243	Mouse component was not detected.	Minor
9246	Mouse component encountered a controller error.	Minor
9266	Local Console component encountered a controller error.	Minor
9268	Local Console component encountered an output error.	Minor
9269	Local Console component encountered a resource conflict error.	Minor
9286	Remote Console component encountered a controller error.	Minor
9287	Remote Console component encountered an input error.	Minor
9288	Remote Console component encountered an output error.	Minor
92A3	Serial port component was not detected	Major
92A9	Serial port component encountered a resource conflict error	Major
92C6	Serial Port controller error	Minor
92C7	Serial Port component encountered an input error.	Minor
92C8	Serial Port component encountered an output error.	Minor
94C6	LPC component encountered a controller error.	Minor
94C9	LPC component encountered a resource conflict error.	Major
9506	ATA/ATPI component encountered a controller error.	Minor
95A6	PCI component encountered a controller error.	Minor
95A7	PCI component encountered a read error.	Minor
95A8	PCI component encountered a write error.	Minor
9609	Unspecified software component encountered a start error.	Minor

Error Code	Error Message	Response
9641	PEI Core component encountered a load error.	Minor
9667	PEI module component encountered an illegal software state error.	Fatal
9687	DXE core component encountered an illegal software state error.	Fatal
96A7	DXE boot services driver component encountered an illegal software state error.	Fatal
96AB	DXE boot services driver component encountered invalid configuration.	Minor
96E7	SMM driver component encountered an illegal software state error.	Fatal
A000	TPM device not detected.	Minor
A001	TPM device missing or not responding.	Minor
A002	TPM device failure.	Minor
A003	TPM device failed self-test.	Minor
A022	Processor component encountered a mismatch error.	Major
A027	Processor component encountered a low voltage error.	Minor
A028	Processor component encountered a high voltage error.	Minor
A100	BIOS ACM Error	Major
A421	PCI component encountered a SERR error.	Fatal
A500	ATA/ATPI ATA bus SMART not supported.	Minor
A501	ATA/ATPI ATA SMART is disabled.	Minor
A5A0	PCI Express component encountered a PERR error.	Minor
A5A1	PCI Express component encountered a SERR error.	Fatal
A5A4	PCI Express IBIST error.	Major
A6A0	DXE boot services driver Not enough memory available to shadow a legacy option ROM.	Minor
B6A3	DXE boot services driver Unrecognized.	Major



## 10. Chassis Subsystem

The BMC monitors several aspects of the chassis. Next to logging when the power and reset buttons get pressed, the BMC also monitors chassis intrusion if a chassis intrusion switch is included in the chassis; as well as looking at the network connections, and logging an event whenever the physical network link is lost.

### 10.1 Physical Security

Two sensors are included in the physical security subsystem: chassis intrusion and LAN leash lost.

#### 10.1.1 Chassis Intrusion

Chassis Intrusion is monitored on supported chassis, and the BMC logs corresponding events when the chassis lid is opened and closed.

#### 10.1.2 LAN Leash Lost

The LAN Leash lost sensor monitors the physical connection on the onboard network ports. If a LAN Leash lost event is logged, this means the network port lost its physical connection.

Table 72: Physical Security Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	05h = Physical Security
12	Sensor Number	04h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as described in Table 73

Byte	Field	Description
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 73: Physical Security Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Chassis intrusion	Somebody has opened the chassis (or the chassis intrusion sensor is not connected).	<ol style="list-style-type: none"> <li>1. Use the <i>Quick Start Guide</i> and the <i>Service Guide</i> to determine whether the chassis intrusion switch is connected properly.</li> <li>2. If this is the case, make sure it makes proper contact when the chassis is closed.</li> <li>3. If this is also the case, someone has opened the chassis. Ensure nobody has access to the system that shouldn't.</li> </ol>
04h	LAN leash lost	Someone has unplugged a LAN cable that was present when the BMC initialized. This event gets logged when the electrical connection on the NIC connector gets lost.	<p>This is most likely due to unplugging the cable but could also happen if there is an issue with the cable or switch.</p> <ol style="list-style-type: none"> <li>1. Check the LAN cable and connector for issues.</li> <li>2. Investigate switch logs where possible.</li> <li>3. Ensure nobody has access to the server that shouldn't.</li> </ol>

## 10.2 FP (NMI) Interrupt

The front panel interrupt button (also referred to as NMI button) is a recessed button on the front panel that allows the user to force a critical interrupt which causes a crash error or kernel panic.

Table 74: FP (NMI) Interrupt Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	13h = Critical Interrupt
12	Sensor Number	05h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 0
15	Event Data 2	Not used
16	Event Data 3	Not used

### 10.2.1 FP (NMI) Interrupt – Next Steps

The purpose of this button is for diagnosing software issues – when a critical interrupt is generated the OS typically saves a memory dump. This allows for exact analysis of what is going on in system memory, which can be useful for software developers, or for troubleshooting OS, software, and driver issues.

If this button was not actually pressed, you should ensure there is no physical fault with the front panel.

This event only gets logged if a user pressed the NMI button, and although it causes the OS to crash, is not an error.

## 10.3 Button Press Events

The BMC logs when the front panel power and reset buttons get pressed. This is purely for informational purposes and these events do not indicate errors.

Table 75: Button Press Events Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	14h = Button/Switch
12	Sensor Number	09h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset 0h = Power Button 2h = Reset Button
15	Event Data 2	Not used
16	Event Data 3	Not used

## 11. Miscellaneous Events

The miscellaneous events section addresses sensors not easily grouped with other sensor types.

### 11.1 IPMI Watchdog

PCSD server systems support an IPMI watchdog timer, which can check to see whether the OS is still responsive. The timer is disabled by default, and has to be enabled manually. It then requires an IPMI-aware utility in the operating system that will reset the timer before it expires. If the timer does expire, the BMC can take action if it is configured to do so (reset, power down, power cycle, or generate a critical interrupt).

Table 76: IPMI Watchdog Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	23h = Watchdog 2
12	Sensor Number	03h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 11B = Sensor-specific event extension code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset as describe in Table 77

Byte	Field	Description
15	Event Data 2	<p>[7:4] – Interrupt type</p> <p>0h = None</p> <p>1h = SMI</p> <p>2h = NMI</p> <p>3h = Messaging Interrupt</p> <p>Fh = Unspecified</p> <p>All other = Reserved</p> <p>[3:0] – Timer use at expiration</p> <p>0h = Reserved</p> <p>1h = BIOS FRB2</p> <p>2h = BIOS/POST</p> <p>3h = OS Load</p> <p>4h = SMS/OS</p> <p>5h = OEM</p> <p>Fh = Unspecified</p> <p>All other = Reserved</p>
16	Event Data 3	Not used

Table 77: IPMI Watchdog Sensor Event Trigger Offset – Next Steps

Event Trigger Offset		Description	Next Steps
Hex	Description		
00h	Timer expired, status only	Our server systems support a BMC watchdog timer, which can check to see whether the OS is still responsive. The timer is disabled by default, and has to be enabled manually. It then requires an IPMI-aware utility in the operating system that will reset the timer before it expires. If the timer does expire, the BMC can take action if it is configured to do so (reset, power down, power cycle, or generate a critical interrupt).	<p>If this event is being logged, it is because the BMC has been configured to check the watchdog timer.</p> <ol style="list-style-type: none"> <li>1. Make sure you have support for this in your OS (typically using a third-party IPMI-aware utility like ipmitool or ipmiutil along with the openipmi driver).</li> <li>2. If this is the case, then it is likely your OS has hung, and you should investigate OS event logs to determine what may have caused this.</li> </ol>
01h	Hard reset		
02h	Power down		
03h	Power cycle		
08h	Timer interrupt		

## 11.2 SMI Timeout

SMI stands for system management interrupt and is an interrupt that gets generated so the processor can service server management events (typically memory or PCI errors, or other forms of critical interrupts), in order to log them to the SEL. If this interrupt times out, the system is frozen.

Table 78: SMI Timeout Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	F3h = SMI Timeout
12	Sensor Number	06h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 03h ("digital" Discrete)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 1 = State Asserted
15	Event Data 2	Not used
16	Event Data 3	Not used

### 11.2.1 SMI Timeout - Next Steps

This event normally only occurs after another more critical event.

1. Check the SEL for any critical interrupts, memory errors, bus errors, PCI errors, or any other serious errors.
2. If these are not present, the system locked up before it was able to log the original issue. In this case, low level debug is normally required.

## 11.3 System Event Log Cleared

The BMC logs a SEL clear event. This is only ever the first event in the SEL. Cause of this event is either a manual SEL clear using Intel® SEL Viewer or some other IPMI-aware utility, or is done in the factory as one of the last steps in the manufacturing process.

This is an informational event only.

Table 79: System Event Log Cleared Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	10h = Event Logging Disabled
12	Sensor Number	07h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 2 = Log area reset/cleared
15	Event Data 2	Not used
16	Event Data 3	Not used

## 11.4 System Event – PEF Action

The BMC is configurable to send alerts for events logged into the SEL. These alerts are called Platform Event Filters (PEF) and are disabled by default. The user must configure and enable this feature. PEF events are logged if the BMC takes action due to a PEF configuration. The BMC event triggering the PEF action will also be in the SEL.



This functionality is built into the BMC to allow it to send alerts (SNMP or other) for any event that gets logged to the SEL. PEF filters are turned off by default and have to be enabled manually using Intel® deployment assistant, Intel® syscfg utility, or an IPMI-aware utility.

Table 80: System Event – PEF Action Sensor Typical Characteristics

Byte	Field	Description
11	Sensor Type	12h = System Event
12	Sensor Number	08h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 11B = Sensor-specific event extension code in Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 4 = PEF Action
15	Event Data 2	[7:6] – Reserved [5] – 1b = Diagnostic Interrupt (NMI) [4] – 1b = OEM action [3] – 1b = Power cycle [2] – 1b = Reset [1] – 1b = Power off [0] – 1b = Alert
16	Event Data 3	Not used

### 11.4.1 System Event – PEF Action – Next Steps

This event gets logged if the BMC takes an action due to PEF configuration. Actions can be sending an alert, or resetting, power cycling, or powering down the system. There will be another event that has led to the action so you should investigate the SEL and PEF settings to identify this event, and troubleshoot accordingly.

## 12. Hot Swap Controller Events

The Hot Swap Controller (HSC) implements the same basic sensor model that is utilized by the other management controllers in the system. Sensor model information is contained in the document *Intelligent Platform Management Interface Specification*. A common set of IPMI commands is used for configuring the sensors and returning threshold status.

### 12.1 HSC Backplane Temperature Sensor

There is a thermal sensor on the Hot Swap Backplane to measure the ambient temperature.

Table 81: HSC Backplane Temperature Sensor Typical Characteristics

Byte	Field	Description
8	Generator ID	00C0h = HSC Firmware – HSBP A
9		00C2h = HSC Firmware – HSBP B
11	Sensor Type	01h = Temperature
12	Sensor Number	01h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 01h (Threshold)
14	Event Data 1	[7:6] – 01b = Trigger reading in Event Data 2 [5:4] – 01b = Trigger threshold in Event Data 3 [3:0] – Event Trigger Offset as described in Table 82
15	Event Data 2	Reading that triggered event
16	Event Data 3	Threshold value that triggered event

Table 82: HSC Backplane Temperature Sensor – Event Trigger Offset – Next Steps

Event Trigger		Assertion Severity	Deassert Severity	Description	Next Steps
Hex	Description				
00h	Lower non-critical going low	Degraded	OK	The temperature has dropped below its lower non-critical threshold.	<ol style="list-style-type: none"> <li>1. Check for clear and unobstructed airflow into and out of the chassis.</li> <li>2. Ensure the SDR is programmed and correct chassis has been selected.</li> <li>3. Ensure there are no fan failures.</li> <li>4. Ensure the air used to cool the system is within the thermal specifications for the system (typically below 35°C).</li> </ol>
02h	Lower critical going low	non-fatal	Degraded	The temperature has dropped below its lower critical threshold.	
07h	Upper non-critical going high	Degraded	OK	The temperature has gone over its upper non-critical threshold.	
09h	Upper critical going high	non-fatal	Degraded	The temperature has gone over its upper critical threshold.	

## 12.2 HSC Drive Slot Status Sensor

The HSC Drive Slot Status sensor provides the current status for drives in each of the slots.

Table 83: HSC Drive Slot Status Sensor Typical Characteristics

Byte	Field	Description	
8	Generator ID	00C0h = HSC Firmware – HSBP A	
9		00C2h = HSC Firmware – HSBP B	
11	Sensor Type	0Dh = Drive Slot (Bay)	
12	Sensor Number	6 Slot HSBP	8 Slot HSBP

Byte	Field	Description	
		02h = Drive Slot 0 Status 03h = Drive Slot 1 Status 04h = Drive Slot 2 Status 05h = Drive Slot 3 Status 06h = Drive Slot 4 Status 07h = Drive Slot 5 Status	02h = Drive Slot 0 Status 03h = Drive Slot 1 Status 04h = Drive Slot 2 Status 05h = Drive Slot 3 Status 06h = Drive Slot 4 Status 07h = Drive Slot 5 Status 08h = Drive Slot 6 Status 09h = Drive Slot 7 Status
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)	
14	Event Data 1	40h = Failed Drive	
15	Event Data 2	Not used	
16	Event Data 3	Not used	

### 12.2.1 HSC Drive Slot Status Sensor – Next Steps

If during normal operation a drive gets reported as failed, ensure that the drive was seated properly and the drive carrier was properly latched. If that does not work, replace the drive.

### 12.3 HSC Drive Presence Sensor

The HSC Drive Slot Presence sensor provides the current presence state for the drive in each of the slots. After an AC power cycle there will be a SEL entry to report the presence of the drive in a slot and there will be another entry for any changes in the presence of drives after that.

Table 84: HSC Drive Presence Sensor Typical Characteristics

Byte	Field	Description	
8 9	Generator ID	00C0h = HSC Firmware – HSBP A 00C2h = HSC Firmware – HSBP B	
11	Sensor Type	0Dh = Drive Slot (Bay)	
12	Sensor Number	6 Slot HSBP	8 Slot HSBP
		08h = Drive Slot 0 Presence 09h = Drive Slot 1 Presence 0Ah = Drive Slot 2 Presence 0Bh = Drive Slot 3 Presence 0Ch = Drive Slot 4 Presence 0Dh = Drive Slot 5 Presence	0Ah = Drive Slot 0 Presence 0Bh = Drive Slot 1 Presence 0Ch = Drive Slot 2 Presence 0Dh = Drive Slot 3 Presence 0Eh = Drive Slot 4 Presence 0Fh = Drive Slot 5 Presence 10h = Drive Slot 6 Presence 11h = Drive Slot 7 Presence
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 08h ("digital" Discrete)	
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset 0h = Device Removed / Device Absent. 1h = Device Inserted / Device Present	
15	Event Data 2	Not used	
16	Event Data 3	Not used	

### 12.3.1 HSC Drive Presence Sensor – Next Steps

On AC power-on the drive presence will be logged as an informational event.

If during normal operation a drive is removed or installed, it will also log an event.

If you get a drive removed or installed without operator intervention, ensure that the drive was seated properly and the drive carrier was properly latched.

## 13. Manageability Engine (ME) Events

The Manageability Engine controls the PECI interface and also contains the Node Manager functionality.

### 13.1 Node Manager Exception Event

A Node Manager Exception Event will be sent each time maintained policy power limit is exceeded over Correction Time Limit.

Table 85: Node Manager Exception Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	002Ch – ME Firmware
11	Sensor Type	DCh = OEM
12	Sensor Number	18h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 72h (OEM)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3] – Node Manager Policy event 0 – Reserved 1 – Policy Correction Time Exceeded – Policy did not meet the contract for the defined policy. The policy will continue to limit the power or shut down the platform based on the defined policy action. [2] – Reserved [1:0] – 00b
15	Event Data 2	[4:7] – Reserved [0:3] – Domain Id (Currently, supports only one domain, Domain 0)
16	Event Data 3	Policy Id

### 13.1.1 Node Manager Exception Event – Next Steps

This is an informational event. Next steps depend on the policy that was set. See the *Node Manager Specification* for more details.

## 13.2 Node Manager Health Event

A Node Manager Health Event message provides a runtime error indication about Intel® Intelligent Power Node Manager's health. Types of service that can send an error are defined as follows:

- Misconfigured policy Error reading power data
- Error reading inlet temperature

Table 86: Node Manager Health Event Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	002Ch – ME Firmware
11	Sensor Type	DCh = OEM
12	Sensor Number	19h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 73h (OEM)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Health Event Type = 02h (Sensor Node Manager)



Byte	Field	Description
15	Event Data 2	<p>[7:4] – Error type</p> <p>0-9 – Reserved</p> <p>10 – Policy Misconfiguration</p> <p>11 – Power Sensor Reading Failure</p> <p>12 – Inlet Temperature Reading Failure</p> <p>13 – Host Communication error</p> <p>14 – Real-time clock synchronization failure</p> <p>15 – Platform shutdown initiated by NM policy due to execution of action defined by Policy Exception Action</p> <p>[3:0] – Domain Id (Currently, supports only one domain, Domain 0)</p>
16	Event Data 3	<p>If Error type = 10 or 15 &lt;Policy Id&gt;</p> <p>If Error type = 11 &lt;Power Sensor Address&gt;</p> <p>If Error type = 12 &lt;Inlet Sensor Address&gt;</p> <p>Otherwise set to 0.</p>

### 13.2.1 Node Manager Health Event – Next Steps

Misconfigured policy can happen if the max/min power consumption of the platform exceeds the values in policy due to hardware reconfiguration.

First occurrence of an unacknowledged event will be retransmitted no faster than every 300 milliseconds.

Real-time clock synchronization failure alert is sent when NM is enabled and capable of limiting power, but within 10 minutes the firmware cannot obtain valid calendar time from the host side, so NM cannot handle suspend periods.

Next steps depend on the policy that was set. See the *Node Manager Specification* for more details.

### 13.3 Node Manager Operational Capabilities Change

This message provides a runtime error indication about Intel® Intelligent Power Node Manager's operational capabilities. This applies to all domains.

Assertion and deassertion of these events are supported.

Table 87: Node Manager Operational Capabilities Change Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	002Ch – ME Firmware
11	Sensor Type	DCh = OEM
12	Sensor Number	1Ah
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 74h (OEM)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Current state of Operational Capabilities. Bit pattern: 0 – Policy interface capability 0 – Not Available 1 – Available 1 – Monitoring capability 0 – Not Available 1 – Available 2 – Power limiting capability 0 – Not Available 1 – Available
15	Event Data 2	Not used

Byte	Field	Description
16	Event Data 3	Not used

### 13.3.1 Node Manager Operational Capabilities Change – Next Steps

Policy Interface available indicates that Intel® Intelligent Power Node Manager is able to respond to the external interface about querying and setting Intel® Intelligent Power Node Manager policies. This is generally available as soon as the microcontroller is initialized.

Monitoring Interface available indicates that Intel® Intelligent Power Node Manager has the capability to monitor power and temperature. This is generally available when firmware is operational.

Power limiting interface available indicates that Intel® Intelligent Power Node Manager can do power limiting and is indicative of an ACPI-compliant OS loaded (unless the OEM has indicated support for non-ACPI compliant OS).

Current value of not acknowledged capability sensor will be retransmitted no faster than every 300 milliseconds.

Next steps depend on the policy that was set. See the *Node Manager Specification* for more details.

## 13.4 Node Manager Alert Threshold Exceeded

Policy Correction Time Exceeded Event will be sent each time maintained policy power limit is exceeded over Correction Time Limit.

Table 88: Node Manager Alert Threshold Exceeded Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	002Ch – ME Firmware
11	Sensor Type	DCh = OEM
12	Sensor Number	1Bh
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 72h (OEM)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3] = Node Manager Policy event 0 – Threshold exceeded 1 – Policy Correction Time Exceeded – Policy did not meet the contract for the defined policy. The policy will continue to limit the power or shut down the platform based on the defined policy action. [2] – Reserved [1:0] – Threshold Number. Valid only if Byte 5 bit [3] is set to 0. 0 to 2 – Threshold index
15	Event Data 2	[7:4] – Reserved [3:0] – Domain Id (Currently, supports only one domain, Domain 0)
16	Event Data 3	Policy ID

### 13.4.1 Node Manager Alert Threshold Exceeded – Next Steps

First occurrence of an unacknowledged event will be retransmitted no faster than every 300 milliseconds.

First occurrence of Threshold exceeded event assertion/deassertion will be retransmitted no faster than every 300 milliseconds.

Next steps depend on the policy that was set. See the *Node Manager Specification* for more details.

## 13.5 ME Firmware Health Event

This sensor is used in Platform Event messages to the BMC containing health information including but not limited to firmware upgrade and application errors.

Table 89: ME Firmware Health Event Sensor Typical Characteristics

Byte	Field	Description
8 9	Generator ID	002Ch or 602Ch – ME Firmware
11	Sensor Type	DCh = OEM
12	Sensor Number	17h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 75h (OEM)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Health event type – 0h (Firmware Status)
15	Event Data 2	See Table 90
16	Event Data 3	See Table 90

### 13.5.1 ME Firmware Health Event – Next Steps

In the following table Event Data 3 is only noted for specific errors.

If the issue continues to be persistent, provide the content of Event Data 3 to Intel support team for interpretation. Event Data 3 codes are in general not documented, because their meaning only provides some clues, varies, and usually needs to be individually interpreted.

Table 90: ME Firmware Health Event Sensor – Next Steps

ED2	ED3	Description	Next Steps
00h		Recovery GPIO forced. Recovery Image loaded due to recovery MGPIOPin asserted. Pin number is configurable in factory presets. Default recovery pin is MGPIOP1.	Deassert MGPIOP1 and reset the Intel® ME.
01h		Image execution failed. Recovery Image or backup operational image loaded because operational image is corrupted. This may be either caused by flash device corruption or failed upgrade procedure.	Either the flash device must be replaced (if error is persistent) or the upgrade procedure must be started again.
02h		Flash erase error. Error during flash erasure procedure.	The flash device must be replaced.
03h		Flash corrupted. Error while checking Flash consistency.	The Flash device must be replaced (if error is persistent).
04h		Internal error. Error during firmware execution – FW Watchdog Timeout.	Operational image needs to be updated to other version or hardware board repair is needed (if error is persistent).
05h		BMC did not respond to cold reset request and Intel® ME rebooted the platform.	Verify the Intel® Node Manager configuration.
06h-FFh		Reserved.	

## 14. Microsoft Windows® Records

With Microsoft Windows Server 2003® R2 and later versions, an Intelligent Platform Management Interface (IPMI) driver was added. This added the capability of logging some OS events to the SEL. The driver can write multiple records to the SEL for the following events:

- Boot-up
- Shutdown
- Bug Check / Blue Screen

### 14.1 Boot-up Event Records

When the system boots into the Microsoft Windows® OS, there can be two events logged. The first is a boot-up record and the second is an OEM event. These are informational only records.

Table 91: Boot-up Event Record Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0041h – System Software with an ID = 20h
11	Sensor Type	1Fh = OS Boot
12	Sensor Number	00h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 1h = C: boot completed
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 92: Boot-up OEM Event Record Typical Characteristics

Byte	Field	Description
1 2	Record ID	ID used for SEL Record access.
3	Record Type	[7:0] – DCh = OEM timestamped, bytes 8-16 OEM defined
4 5 6 7	Timestamp	Time when event was logged. LS byte first.
8 9 10	IPMI Manufacturer ID	0137h (311d) = IANA enterprise number for Microsoft
11	Record ID	Sequential number reflecting the order in which the records are read. The numbers start at 1 for the first entry in the SEL and continue sequentially to <i>n</i> , the number of entries in the SEL.
12 13 14 15	Boot Time	Timestamp of when system booted into the OS
16	Reserved	00h

## 14.2 Shutdown Event Records

When the system shuts down from the Microsoft Windows\* OS, there can be multiple events logged. The first is an OS Stop/Shutdown Event Record; this can be followed by a shutdown reason code OEM record, and then zero or more shutdown comment OEM records. These are all informational only records.



Table 93: Shutdown Reason Code Event Record Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0041h – System Software with an ID = 20h
11	Sensor Type	20h = OS Stop/Shutdown
12	Sensor Number	00h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 3h = OS Graceful Shutdown
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 94: Shutdown Reason OEM Event Record Typical Characteristics

Byte	Field	Description
1 2	Record ID	ID used for SEL Record access.
3	Record Type	[7:0] – DDh = OEM timestamped, bytes 8-16 OEM defined
4 5 6 7	Timestamp	Time when event was logged. LS byte first.
8 9 10	IPMI Manufacturer ID	0137h (311d) = IANA enterprise number for Microsoft

Byte	Field	Description
11	Record ID	Sequential number reflecting the order in which the records are read. The numbers start at 1 for the first entry in the SEL and continue sequentially to <i>n</i> , the number of entries in the SEL.
12 13 14 15	Shutdown Reason	Shutdown Reason code from the registry (LSB first.): HKLM/Software/Microsoft/Windows/CurrentVersion/Reliability/shutdown/ReasonCode
16	Reserved	00h

Table 95: Shutdown Comment OEM Event Record Typical Characteristics

Byte	Field	Description
1 2	Record ID	ID used for SEL Record access.
3	Record Type	[7:0] – DDh = OEM timestamped, bytes 8-16 OEM defined
4 5 6 7	Timestamp	Time when event was logged. LS byte first.
8 9 10	IPMI Manufacturer ID	0137h (311d) = IANA enterprise number for Microsoft 0157h (343) = IANA enterprise number for Intel The value logged depends on the Intelligent Management Bus Driver (IMBDRV) that is loaded.
11	Record ID	Sequential number reflecting the order in which the records are read. The numbers start at 1 for the first entry in the SEL and continue sequentially to <i>n</i> , the number of entries in the SEL.
12 13 14 15	Shutdown Comment	Shutdown Comment from the registry (LSB first.): HKLM/Software/Microsoft/Windows/CurrentVersion/Reliability/shutdown/Comment
16	Reserved	00h

## 14.3 Bug Check / Blue Screen Event Records

When the system experiences a bug check (blue screen), there will be multiple records written to the event log. The first is a Bug Check / Blue Screen OS Stop/Shutdown Event Record; this can be followed by multiple Bug Check / Blue Screen code OEM records that will contain the Bug Check / Blue Screen codes. This information can be used to determine what caused the failure.

Table 96: Bug Check / Blue Screen – OS Stop Event Record Typical Characteristics

Byte	Field	Description
8 9	Generator ID	0041h – System Software with an ID = 20h
11	Sensor Type	20h = OS Stop/Shutdown
12	Sensor Number	00h
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 00b = Unspecified Event Data 2 [5:4] – 00b = Unspecified Event Data 3 [3:0] – Event Trigger Offset = 1h = Runtime Critical Stop (that is, “core dump”, “blue screen”)
15	Event Data 2	Not used
16	Event Data 3	Not used

Table 97: Bug Check / Blue Screen Code OEM Event Record Typical Characteristics

Byte	Field	Description
1 2	Record ID	ID used for SEL Record access.
3	Record Type	[7:0] – DEh = OEM timestamped, bytes 8-16 OEM defined

Byte	Field	Description
4 5 6 7	Timestamp	Time when event was logged. LS byte first.
8 9 10	IPMI Manufacturer ID	0137h (311) = IANA enterprise number for Microsoft 0157h (343) = IANA enterprise number for Intel The value logged depends on the Intelligent Management Bus Driver (IMBDRV) that is loaded.
11	Sequence Number	Sequential number reflecting the order in which the records are read. The numbers start at 1 for the first entry in the SEL and continue sequentially to $n$ , the number of entries in the SEL.
12 13 14 15	Bug Check/Blue Screen Data	The first record of this type will contain the Bug Check / Blue Screen Stop code and will be followed by the four Bug Check / Blue Screen parameters. LSB first. Note that each of the Bug Check / Blue Screen parameters requires two records each. Both of the two records for each parameter will have the same Record ID. There will be a total of 9 records.
16	Operating system type	00 = 32 bit OS 01 = 64 bit OS

## 15. Linux\* Kernel Panic Records

The OpenIPMI driver supports the ability to put semi-custom and custom events in the system event log if a panic occurs. If you enable the “Generate a panic event to all BMCs on a panic” option, you will get one event on a panic in a standard IPMI event format. If you enable the “Generate OEM events containing the panic string” option, you will also get a set of OEM events holding the panic string.

Table 98: Linux\* Kernel Panic Event Record Characteristics

Byte	Field	Description
8 9	Generator ID	0021h – Kernel
10	EvM Rev	03h = IPMI 1.0 format
11	Sensor Type	20h = OS Stop/Shutdown
12	Sensor Number	The first byte of the panic string (0 if no panic string)
13	Event Direction and Event Type	[7] Event direction 0b = Assertion Event 1b = Deassertion Event [6:0] Event Type = 6Fh (Sensor Specific)
14	Event Data 1	[7:6] – 10b = OEM code in Event Data 2 [5:4] – 10b = OEM code in Event Data 3 [3:0] – Event Trigger Offset = 1h = Runtime Critical Stop (that is, “core dump”, “blue screen”)
15	Event Data 2	The second byte of panic string
16	Event Data 3	The third byte of panic string

Table 99: Linux\* Kernel Panic String Extended Record Characteristics

Byte	Field	Description
1 2	Record ID	ID used for SEL Record access.
3	Record Type	[7:0] – F0h = OEM non-timestamped, bytes 4-16 OEM defined
4	Slave Address	The slave address of the card saving the panic.
5	Sequence Number	A sequence number (starting at zero).
6 ... 16	Kernel Panic Data	These hold the panic sting. If the panic string is longer than 11 bytes, multiple messages will be sent with increasing sequence numbers.